

日本国特許庁
JAPAN PATENT OFFICE

REC'D 03 MAR 2005
WIPO PCT

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出願年月日 2004年 2月 20日
Date of Application:

出願番号 特願 2004-044649
Application Number:
[ST. 10/C] : [JP 2004-044649]

出願人 セイコーエプソン株式会社
Applicant(s):

PRIORITY DOCUMENT
SUBMITTED OR TRANSMITTED IN
COMPLIANCE WITH
RULE 17.1(a) OR (b)

2004年12月 8日

特許庁長官
Commissioner,
Japan Patent Office

小川

洋

【書類名】 特許願
【整理番号】 J0106806
【あて先】 特許庁長官 殿
【国際特許分類】 H04N 1/60
【発明者】
【住所又は居所】 長野県諏訪市大和三丁目 3 番 5 号 セイコーエプソン株式会社内
【氏名】 角谷 繁明
【特許出願人】
【識別番号】 000002369
【氏名又は名称】 セイコーエプソン株式会社
【代理人】
【識別番号】 100095728
【弁理士】
【氏名又は名称】 上柳 雅誉
【連絡先】 0266-52-3528
【選任した代理人】
【識別番号】 100107076
【弁理士】
【氏名又は名称】 藤綱 英吉
【選任した代理人】
【識別番号】 100107261
【弁理士】
【氏名又は名称】 須澤 修
【手数料の表示】
【予納台帳番号】 013044
【納付金額】 21,000円
【提出物件の目録】
【物件名】 特許請求の範囲 1
【物件名】 明細書 1
【物件名】 図面 1
【物件名】 要約書 1
【包括委任状番号】 0109826

【書類名】特許請求の範囲

【請求項1】

画像データに所定の画像処理を施す画像処理装置と、該画像処理の結果に基づいてドットを形成することにより出力媒体上に画像を出力する画像出力装置とを備える画像出力システムであって、

前記画像処理装置は、

互いに隣接する画素を所定個数ずつ画素群にまとめて前記画像を複数の画素群に分割し、該各々の画素群内に形成されるドットの個数を表す個数データを、前記画像データに基づいて生成する個数データ生成手段と、

前記画素群毎に生成された個数データを、前記画像出力装置に供給する個数データ供給手段と

を備えており、

前記画像出力装置は、

前記画素群内の各画素について、該画素群内でドットが形成される順番を示す順序値を取得する順序値取得手段と、

前記順序値および前記個数データの組合せと、該順序値を有する画素におけるドット形成の有無との対応関係を記憶している対応関係記憶手段と、

前記画素群についての個数データを受け取ると、該個数データと前記各々の順序値との組合せ毎に前記対応関係を参照することにより、該画素群内の各画素についてのドット形成の有無を決定するドット形成有無決定手段と、

前記決定したドット形成の有無に従って前記出力媒体上にドットを形成するドット形成手段と

を備えている画像出力システム。

【請求項2】

請求項1記載の画像出力システムであって、

前記画素群内でドットが形成される画素の序列を複数組記憶している序列記憶手段を備えるとともに、

前記順序値取得手段は、前記複数組の序列の中から前記画素群毎に選択した1の序列に基づいて、該画素群内の各画素についての前記順序値を取得する手段であり、

前記ドット形成有無決定手段は、前記取得した順序値を用いて、前記画素群内の各画素についてのドット形成の有無を決定する手段である画像出力システム。

【請求項3】

請求項2記載の画像出力システムであって、

前記個数データ生成手段は、二次元的に配列された画素の各々に閾値を対応付けたディザマトリックスに基づいて、前記個数データを生成する手段であり、

前記序列記憶手段は、前記個数データの生成に用いたディザマトリックスを複数の画素群に分割し、該画素群内の各画素に対応付けられた閾値の大小関係に基づいて画素群毎に決定された画素の序列を、前記複数組の画素の序列として記憶している手段である画像出力システム。

【請求項4】

所定の画像処理が施された画像データを受け取って、該画像データに基づいて出力媒体上にドットを形成することにより、画像を出力する画像出力装置であって、

互いに隣接する画素が所定個数ずつ画素群としてまとめられた状態で、該画素群内に形成すべきドットの個数を表す個数データを、前記画像データとして受け取る個数データ受取手段と、

前記画素群内の各画素について、該画素群内でドットが形成される順番を示す順序値を取得する順序値取得手段と、

前記順序値および前記個数データの組合せと、該順序値を有する画素におけるドット形成の有無との対応関係を記憶している対応関係記憶手段と、

前記個数データを受け取った画素群内の各画素について、該個数データと前記順序値と

の組合せ毎に前記対応関係を参照することにより、ドット形成の有無を決定するドット形成有無決定手段と、

前記決定したドット形成の有無に従って前記出力媒体上にドットを形成するドット形成手段と

を備える画像出力装置。

【請求項5】

請求項4記載の画像出力装置であって、

前記画素群内でドットが形成される画素の序列を複数組記憶している序列記憶手段を備えるとともに、

前記順序値取得手段は、前記複数組の序列の中から前記画素群毎に選択した1の序列に基づいて、該画素群内の各画素についての前記順序値を取得する手段であり、

前記ドット形成有無決定手段は、前記取得した順序値を用いて、前記画素群内の各画素についてのドット形成の有無を決定する手段である画像出力装置。

【請求項6】

請求項4または請求項5記載の画像出力装置であって、

前記個数データ受取手段は、表現する階調値の異なる複数種類のドットについて、前記画素群内に形成される各種ドットの個数を、前記個数データとして受け取る手段であり、

前記対応関係記憶手段は、前記順序値および前記個数データの組合せと、該順序値を有する画素に形成されるドット種類との対応関係を記憶している手段であり、

前記ドット形成有無決定手段は、前記画素群内の各画素について、前記各種ドットについての形成の有無を決定する手段であり、

前記ドット形成手段は、前記決定したドット形成の有無に従って、前記出力媒体上に前記各種ドットを形成する手段である画像出力装置。

【請求項7】

請求項4ないし請求項6のいずれかに記載の画像出力装置であって、

前記個数データ受取手段は、互いに所定の位置関係にある8画素ないし16画素ずつの画素がまとめられた画素群について、前記個数データを受け取る手段である画像出力装置。

【請求項8】

画像データに応じて出力媒体上にドットを形成することにより、該画像データに対応した画像を出力する画像出力装置であって、

互いに隣接する画素を所定個数ずつ画素群にまとめて前記画像を複数の画素群に分割し、該各々の画素群内に形成されるドットの個数を表す個数データを、前記画像データに基づいて生成する個数データ生成手段と、

前記画素群内の各画素について、該画素群内でドットが形成される順番を示す順序値を取得する順序値取得手段と、

前記順序値および前記個数データの組合せと、該順序値を有する画素におけるドット形成の有無との対応関係を記憶している対応関係記憶手段と、

前記個数データを生成した画素群内の各画素について、該個数データと前記順序値との組合せ毎に前記対応関係を参照することにより、ドット形成の有無を決定するドット形成有無決定手段と、

前記決定したドット形成の有無に従って前記出力媒体上にドットを形成するドット形成手段と

を備える画像出力装置。

【請求項9】

画像データに所定の画像処理を施して、得られた結果に基づいてドットを形成することにより、出力媒体上に画像を出力する画像出力方法であって、

互いに隣接する画素を所定個数ずつ画素群にまとめて前記画像を複数の画素群に分割し、該各々の画素群内に形成されるドットの個数を表す個数データを、前記画像データに基づいて生成する第1の工程と、

前記画素群内の各画素について、該画素群内でドットが形成される順番を示す順序値を取得する第2の工程と、

前記順序値および前記個数データの組合せと、該順序値を有する画素におけるドット形成の有無との対応関係を記憶しておく第3の工程と、

前記個数データを生成した画素群内の各画素について、該個数データと前記順序値との組合せ毎に前記対応関係を参照することにより、ドット形成の有無を決定する第4の工程と、

前記決定したドット形成の有無に従って前記出力媒体上にドットを形成する第5の工程と

を備える画像出力方法。

【請求項10】

所定の画像処理が施された画像データを受け取って、該画像データに基づいて出力媒体上にドットを形成することにより、画像を出力する画像出力方法であって、

互いに隣接する画素が所定個数ずつ画素群としてまとめられた状態で、該画素群内に形成すべきドットの個数を表す個数データを、前記画像データとして受け取る工程（A）と、

前記画素群内の各画素について、該画素群内でドットが形成される順番を示す順序値を取得する工程（B）と、

前記順序値および前記個数データの組合せと、該順序値を有する画素におけるドット形成の有無との対応関係を記憶しておく工程（C）と、

前記個数データを受け取った画素群内の各画素について、該個数データと前記順序値との組合せ毎に前記対応関係を参照することにより、ドット形成の有無を決定する工程（D）と、

前記決定したドット形成の有無に従って前記出力媒体上にドットを形成する工程（E）と

を備える画像出力方法。

【請求項11】

画像データに所定の画像処理を施して、得られた結果に基づいてドットを形成することにより出力媒体上に画像を出力する方法を、コンピュータを用いて実現するためのプログラムであって、

互いに隣接する画素を所定個数ずつ画素群にまとめて前記画像を複数の画素群に分割し、該各々の画素群内に形成されるドットの個数を表す個数データを、前記画像データに基づいて生成する第1の機能と、

前記画素群内の各画素について、該画素群内でドットが形成される順番を示す順序値を取得する第2の機能と、

前記順序値および前記個数データの組合せと、該順序値を有する画素におけるドット形成の有無との対応関係を記憶しておく第3の機能と、

前記個数データを生成した画素群内の各画素について、該個数データと前記順序値との組合せ毎に前記対応関係を参照することにより、ドット形成の有無を決定する第4の機能と、

前記決定したドット形成の有無に従って前記出力媒体上にドットを形成する第5の機能と

を実現するプログラム。

【請求項12】

所定の画像処理が施された画像データを受け取って、該画像データに基づいて出力媒体上にドットを形成することにより画像を出力する方法を、コンピュータを用いて実現するためのプログラムであって、

互いに隣接する画素が所定個数ずつ画素群としてまとめられた状態で、該画素群内に形成すべきドットの個数を表す個数データを、前記画像データとして受け取る機能（A）と、

前記画素群内の各画素について、該画素群内でドットが形成される順番を示す順序値を取得する機能（B）と、

前記順序値および前記個数データの組合せと、該順序値を有する画素におけるドット形成の有無との対応関係を記憶しておく機能（C）と、

前記個数データを受け取った画素群内の各画素について、該個数データと前記順序値との組合せ毎に前記対応関係を参照することにより、ドット形成の有無を決定する機能（D）と、

前記決定したドット形成の有無に従って前記出力媒体上にドットを形成する機能（E）と

を実現するプログラム。

【書類名】明細書

【発明の名称】所定領域内に形成されるドット個数の情報に基づいて画像を出力する画像出力システム

【技術分野】

【0001】

この発明は、画像データに基づいて画像を出力する技術に関し、詳しくは、画像データに所定の画像処理を施してドットを適切な密度で発生させることにより、画像を出力する技術に関する。

【背景技術】

【0002】

印刷媒体や液晶画面といった各種の出力媒体上にドットを形成することで画像を出力する画像出力装置は、各種画像機器の出力装置として広く使用されている。これら画像出力装置では、画像は画素と呼ばれる小さな領域に細分された状態で扱われており、ドットはこれら画素に形成される。ドットを画素に形成した場合、もちろん画素1つ1つについて見れば、ドットが形成されるか否かのいずれかの状態しか取り得ない。しかし、ある程度の広さを持った領域で見れば、ドットの形成され方に粗密を生じさせることは可能であり、ドットの形成密度を変えることによって多階調の画像を出力することが可能である。例えば、印刷用紙上に黒いインクのドットを形成する場合、ドットが密に形成されている領域は暗く見えるし、逆にドットがまばらに形成されている領域は明るく見える。また、液晶画面に輝点のドットを形成する場合、ドットが密に形成された領域は明るく見え、まばらに形成された領域は暗く見える。従って、ドットの形成密度を適切に制御してやれば、多階調の画像を出力することが可能となる。このように、適切な形成密度が得られるようにドットの形成を制御するためのデータは、出力しようとする画像に所定の画像処理を施すことによって発生させる。

【0003】

近年では、これら画像出力装置には、出力画像の高画質化や大画像化が要請されるようになってきた。高画質化の要請に対しては、画像をより細かな画素に分割することが効果的である。画素を小さくしてやれば、画素に形成されるドットが目立たなくなるので画質を向上させることができる。また、大画像化の要請に対しては、画素数の増加によって対応する。もちろん、個々の画素を大きくすることによっても出力画像を大きくすることはできるが、これでは画質の低下を招いてしまうので、大型化の要請に対しては画素数を増加させることが効果的である。

【0004】

もっとも、画像を構成する画素数が増加すると画像処理に時間がかかってしまい、画像を迅速に出力することが困難となる。そこで、画像処理を迅速に実行可能とする技術が提案されている（特許文献1）。

【0005】

【特許文献1】特開2002-185789号公報

【発明の開示】

【発明が解決しようとする課題】

【0006】

しかし、画像処理を迅速に行ったとしても、画像データの転送や転送されたデータの処理の長い時間がかかってしまったのでは、画像を迅速に出力する効果には自ずから限界が生じる。

【0007】

この発明は従来技術における上述した課題を解決するためになされたものであり、十分な出力画質を維持したまま、画像データの転送や転送されたデータを高速に処理可能な技術の提供を目的とする。

【課題を解決するための手段】

【0008】

上述した課題の少なくとも一部を解決するために、本発明の画像出力システムは、次の構成を採用した。すなわち、

画像データに所定の画像処理を施す画像処理装置と、該画像処理の結果に基づいてドットを形成することにより出力媒体上に画像を出力する画像出力装置とを備える画像出力システムであって、

前記画像処理装置は、

互いに隣接する画素を所定個数ずつ画素群にまとめて前記画像を複数の画素群に分割し、該各々の画素群内に形成されるドットの個数を表す個数データを、前記画像データに基づいて生成する個数データ生成手段と、

前記画素群毎に生成された個数データを、前記画像出力装置に供給する個数データ供給手段と

を備えており、

前記画像出力装置は、

前記画素群内の各画素について、該画素群内でドットが形成される順番を示す順序値を取得する順序値取得手段と、

前記順序値および前記個数データの組合せと、該順序値を有する画素におけるドット形成の有無との対応関係を記憶している対応関係記憶手段と、

前記画素群についての個数データを受け取ると、該個数データと前記各々の順序値との組合せ毎に前記対応関係を参照することにより、該画素群内の各画素についてのドット形成の有無を決定するドット形成有無決定手段と、

前記決定したドット形成の有無に従って前記出力媒体上にドットを形成するドット形成手段と

を備えることを要旨とする。

【0009】

また、上記の画像出力システムに対応する本発明の画像出力方法は、

画像データに所定の画像処理を施して、得られた結果に基づいてドットを形成することにより、出力媒体上に画像を出力する画像出力方法であって、

互いに隣接する画素を所定個数ずつ画素群にまとめて前記画像を複数の画素群に分割し、該各々の画素群内に形成されるドットの個数を表す個数データを、前記画像データに基づいて生成する第1の工程と、

前記画素群内の各画素について、該画素群内でドットが形成される順番を示す順序値を取得する第2の工程と、

前記順序値および前記個数データの組合せと、該順序値を有する画素におけるドット形成の有無との対応関係を記憶しておく第3の工程と、

前記個数データを生成した画素群内の各画素について、該個数データと前記順序値との組合せ毎に前記対応関係を参照することにより、ドット形成の有無を決定する第4の工程と、

前記決定したドット形成の有無に従って前記出力媒体上にドットを形成する第5の工程と

を備えることを要旨とする。

【0010】

かかる本発明の画像出力システムおよび画像出力方法においては、画像を複数の画素群に分割し、画素群毎に個数データを生成して、画像出力装置に供給する。画像出力装置では、画素群内の各画素についてドットが形成される順番を示す順序値を取得する。各画素の順序値を取得するに際しては、例えば、画素群内の各画素に「1」から始まる連続した整数値を予め設定しておき、この整数値を読み出して順序値と読み替えて良いし、あるいは、各画素に異なる値の実数値を予め設定しておき、実数値の大きさの順に従って、各画素の順序値を決定しても良い。更には、各画素間に前後関係を設定しておき、これら前後関係に基づいて各画素の順序値を決定しても良い。また、画像出力装置には、順序値および個数データの組合せと、該順序値を有する画素におけるドット形成の有無との対応関

係も記憶されている。そして、画素群についての個数データを受け取ると、該個数データと該画素群内の各画素についての順序値との組合せから該対応関係を参照することによって、画素群内の各画素についてのドット形成の有無を決定する。こうして決定したドット形成の有無に従って、出力媒体上にドットを形成することによって画像を出力する。

【0011】

詳細には後述するが、画像の全画素についてドット形成の有無を表すデータに比べれば、画素群毎にドットの個数を表すデータは遙かに小さなデータとすることができます。このため、画素群毎の個数データを画像出力装置に供給してやれば、迅速にデータを供給することが可能となり、延いては画像を迅速に出力することが可能となる。

【0012】

また、画像出力装置は画素群毎に出力された個数データを受け取ると、個数データと画素群内の各画素についての順序値との組合せから対応関係を参照することによって、画素群内の各画素についてのドット形成の有無を決定している。このように対応関係を参照しながらドット形成の有無を決定してやれば、画素群内の各画素についてのドット形成の有無を、個数データから直ちに決定することができるので、極めて迅速に且つ簡便に決定可能であり、延いては画像を迅速に出力することが可能となる。

【0013】

更に、このように極めて簡素な処理でドット形成の有無を決定することができれば、画像出力装置が画像処理装置のようには高度な処理能力を有さない場合でも、個数データから各画素についてのドット形成有無を迅速に決定することが可能である。

【0014】

こうした画像出力システムにおいては、各画素の順序値を次のようにして取得して、ドット形成の有無を決定しても良い。先ず、画素群内でドットが形成される画素の序列を複数組記憶しておき、これら複数組の序列の中から画素群毎に1の序列を選択する。そして、選択した序列に基づいて、該画素群内の各画素についての順序値を取得し、得られた順序値に基づいて各画素のドット形成の有無を決定してもよい。

【0015】

画素群内の各画素についてのドット形成有無は、個数データと順序値に基づいて決定される。従って、複数組の序列の中から1の序列を選択して、その序列に基づいて取得された順序値を用いてドット形成の有無を決定してやれば、例え、同じ個数データが続いた場合でも、複数の画素群に亘って同じ画素位置にドットが形成されることがない。このため、同じパターンでドットが形成された領域が目立ってしまい、画質が悪化することを確実に回避することが可能となる。

【0016】

また、こうした画像出力システムにおいては、二次元的に配列された画素の各々に閾値を対応付けたディザマトリックスに基づいて個数データを生成するとともに、同じディザマトリックスに基づいて取得された順序値を用いて、各画素のドット形成の有無を決定することとしても良い。すなわち、ディザマトリックスを複数の画素群に分割し、画素群内に形成されるドット個数を求めて個数データを生成する。個数データの生成に際しては、分割されたディザマトリックスを用いて、画素群内の各画素についてドット形成の有無を判断することにより、画素群内に形成されるドット個数のデータを生成すればよい。あるいは、個数データを生成するためには、ドットが形成される画素位置までは分からなくても良いことから、簡易的には次のようにすることもできる。先ず、分割されたディザマトリックスに設定されている閾値のみを画素群毎に記憶しておく。次いで、画素群を代表する階調値（代表階調値）を決定する。代表階調値には、各画素の画像データの平均値を用いることできるし、あるいは、画像データは隣接する画素間では近似した値を有することから、画素群内の所定位置の画素の画像データを代表階調値とすることも可能である。そして、代表階調値より小さな閾値の個数を画素群毎に求めて、得られた値をその画素群のドット個数とすることもできる。

【0017】

このようにして生成した個数データを受け取ると、画素群内の各画素についてのドット形成の有無を、次のようにして決定する。予め、個数データの生成に用いたディザマトリックスを複数の画素群に分割し、画素群内の各画素に対応付けられた閾値に基づいて決定された画素の序列を、複数組記憶しておく。あるいは、画素群内の各画素に対応付けられた閾値の大小関係に基づいて画素群毎に各画素の順序値を決定し、得られた順序値を画素の序列として複数組記憶しておく。そして、画素群の個数データを受け取ると、画像上で該画素群の位置に対応した1の序列を選択して、該序列に基づいて各画素の順序値を取得した後、ドット形成の有無を決定することとしてもよい。

【0018】

詳細には後述するが、このようにして、ディザマトリックスに基づいて画素群の個数データを生成し、該画素群内の各画素についてのドット形成の有無を、同じディザマトリックスに基づいて決定してやれば、ディザ法を用いて画素毎にドット形成の有無を判断した場合と同等の画質で画像を出力することが可能となる。特に、画素群としてまとめられる画素の画像データが同じ階調値を有している場合は、ディザ法を用いて画素毎にドット形成の有無を判断した場合と、個数データからドットを形成する画素位置を決定した場合とで、全く同じ画素位置にドットが形成されることになる。

【0019】

また、従来技術が有する前述した課題の少なくとも一部を解決するために、本発明の第1の画像出力装置は次の構成を採用した。すなわち、

所定の画像処理が施された画像データを受け取って、該画像データに基づいて出力媒体上にドットを形成することにより、画像を出力する画像出力装置であって、

互いに隣接する画素が所定個数ずつ画素群としてまとめられた状態で、該画素群内に形成すべきドットの個数を表す個数データを、前記画像データとして受け取る個数データ受取手段と、

前記画素群内の各画素について、該画素群内でドットが形成される順番を示す順序値を取得する順序値取得手段と、

前記順序値および前記個数データの組合せと、該順序値を有する画素におけるドット形成の有無との対応関係を記憶している対応関係記憶手段と、

前記個数データを受け取った画素群内の各画素について、該個数データと前記順序値との組合せ毎に前記対応関係を参照することにより、ドット形成の有無を決定するドット形成有無決定手段と、

前記決定したドット形成の有無に従って前記出力媒体上にドットを形成するドット形成手段と

を備えることを要旨とする。

【0020】

また、上記の第1の画像出力装置に対応する本発明の画像出力方法は、

所定の画像処理が施された画像データを受け取って、該画像データに基づいて出力媒体上にドットを形成することにより、画像を出力する画像出力方法であって、

互いに隣接する画素が所定個数ずつ画素群としてまとめられた状態で、該画素群内に形成すべきドットの個数を表す個数データを、前記画像データとして受け取る工程（A）と

前記画素群内の各画素について、該画素群内でドットが形成される順番を示す順序値を取得する工程（B）と、

前記順序値および前記個数データの組合せと、該順序値を有する画素におけるドット形成の有無との対応関係を記憶しておく工程（C）と、

前記個数データを受け取った画素群内の各画素について、該個数データと前記順序値との組合せ毎に前記対応関係を参照することにより、ドット形成の有無を決定する工程（D）と、

前記決定したドット形成の有無に従って前記出力媒体上にドットを形成する工程（E）と

を備えることを要旨とする。

【0021】

かかる画像出力装置および画像出力方法においては、画素群の個数データを受け取ると、順序値および個数データの組合せと、該順序値を有する画素におけるドット形成の有無との対応関係を参照しながら、画素群内の各画素についてのドット形成の有無を決定する。こうして決定したドット形成の有無に従って、出力媒体上にドットを形成することによって画像を出力する。

【0022】

後述するように、画素群の個数データであれば迅速にデータを受け取ることができるの、画像を迅速に出力することが可能である。また、各画素についてのドット形成の有無は、個数データおよび順序値の組合せ毎に設定された対応関係を参照することによって決定することができるので、簡便に且つ迅速に決定することができる。このため、画像を迅速に出力することが可能であるとともに、高度な処理能力を有さない画像出力装置においても、十分に実用的な速度で画像を出力することが可能となる。

【0023】

こうした画像出力装置においては、画素群内でドットが形成される画素の序列を複数組記憶しているとともに、該画素の序列毎に各画素についての前記順序値を記憶しておき、個数データを受け取ると、これら複数の序列の中から画素群毎に1の序列を選択し、該序列に基づいて取得された順序値を用いて各画素のドット形成の有無を決定することとしてもよい。

【0024】

こうすれば、同じ個数データが連続した場合でも、複数の画素群に亘って同じ画素位置にドットが形成されることがないので、同じパターンでドットが形成された領域が目立つてしまい、画質が悪化することを回避することが可能となる。

【0025】

このような画像出力装置においては、表現する階調値の異なる複数種類のドットを出力可能として、画素群内に形成される各種ドットの個数を、個数データとして受け取ることとしてもよい。ここで、表現する階調値の異なる複数種類のドットとは、例えば、ドットの大きさが異なる複数種類のドットとすることもできるし、あるいは、ドットの濃度が異なる複数種類のドットとすることもできる。更には、微細なドットを所定の密度で形成することで擬似的に1つのドットを形成している場合には、微細なドットの密度が異なった複数種類のドットとすることも可能である。これら各種ドットを形成可能な場合は、各種ドットの個数の組合せを個数データとして受け取る。そして、順序値および前記個数データの組合せと、該順序値を有する画素に形成されるドット種類との対応関係を記憶しておき、個数データを受け取ると、該対応関係を参照することによって、各画素に形成されるドットの種類を決定し、決定したドット形成の有無に従って、出力媒体上に各種ドットを形成することとしてもよい。

【0026】

画素群内の各画素についてドット形成有無を、対応関係を参照しながら決定する場合は、個数データが複数種類のドットについてのドット個数の組合せを示すデータであっても、各画素についてのドット形成の有無を、単にドットの個数を表すデータである場合と同様に、極めて容易に決定することができる。このため、ドット形成有無を迅速に決定することができ、延いては画像を迅速に出力することが可能となるので好ましい。

【0027】

こうした画像出力装置においては、互いに所定の位置関係にある8画素ないし16画素ずつの画素がまとめられた画素群について、個数データを受け取ることとしてもよい。

【0028】

画素群としてまとめる画素数が多くなるほど画素群の数は少なくなるので、個数データを迅速に受け取ることができる。一方、画素群にまとめる画素数があまりに多くなったのでは画質の悪化を引き起こすおそれがある。こうした点を踏まえると、経験上、8画素

から16画素ずつの画素を画素群としてまとめた場合に、最も良好な結果を得ることができる。すなわち、詳細には後述するが、画素群にまとめられる画素数を8画素から16画素にしておけば、個数データのデータ量を、画素毎にドット形成の有無を表したデータの半分以下に低減することができ、迅速にデータを受け取ることができる。また、画素群にまとめられる複数の画素の位置関係は、例えば、主走査方向に4画素ずつ、副走査方向に2画素ずつといったように、互いに矩形形状を成すような位置関係としておけば、経験上、良好な画質を得ることが可能である。

【0029】

また、従来技術が有する前述した課題の少なくとも一部を解決するために、本発明の第2の画像出力装置は次の構成を採用した。すなわち、

画像データに応じて出力媒体上にドットを形成することにより、該画像データに対応した画像を出力する画像出力装置であって、

互いに隣接する画素を所定個数ずつ画素群にまとめて前記画像を複数の画素群に分割し、該各々の画素群内に形成されるドットの個数を表す個数データを、前記画像データに基づいて生成する個数データ生成手段と、

前記画素群内の各画素について、該画素群内でドットが形成される順番を示す順序値を取得する順序値取得手段と、

前記順序値および前記個数データの組合せと、該順序値を有する画素におけるドット形成の有無との対応関係を記憶している対応関係記憶手段と、

前記個数データを生成した画素群内の各画素について、該個数データと前記順序値との組合せ毎に前記対応関係を参照することにより、ドット形成の有無を決定するドット形成有無決定手段と、

前記決定したドット形成の有無に従って前記出力媒体上にドットを形成するドット形成手段と

を備えることを要旨とする。

【0030】

かかる本発明の画像出力装置においても、画像を複数の画素群に分割し、画素群内に形成されるドット個数を表す個数データを生成する。次いで、順序値および個数データの組合せと、該順序値を有する画素におけるドット形成の有無との対応関係を参照しながら、画素群内の各画素についてのドット形成の有無を決定する。こうして決定したドット形成の有無に従って、出力媒体上にドットを形成することによって画像を出力する。

【0031】

各画素についてのドット形成の有無を、このように個数データおよび順序値の組合せ毎に設定された対応関係を参照しながら決定してやれば、簡便に且つ迅速に決定することができる。このため、画像を迅速に出力することが可能であるとともに、高度な処理能力を有さない画像出力装置においても、十分に実用的な速度で画像を出力することが可能となる。

【0032】

更に本発明は、上述した画像出力方法を実現するためのプログラムをコンピュータに読み込ませ、コンピュータを用いて実現することも可能である。従って、本発明は次のようなプログラム、あるいは該プログラムを記録した記録媒体としての態様も含んでいる。すなわち、上述した画像出力方法に対応する本発明の第1のプログラムは、

画像データに所定の画像処理を施して、得られた結果に基づいてドットを形成することにより出力媒体上に画像を出力する方法を、コンピュータを用いて実現するためのプログラムであって、

互いに隣接する画素を所定個数ずつ画素群にまとめて前記画像を複数の画素群に分割し、該各々の画素群内に形成されるドットの個数を表す個数データを、前記画像データに基づいて生成する第1の機能と、

前記画素群内の各画素について、該画素群内でドットが形成される順番を示す順序値を取得する第2の機能と、

前記順序値および前記個数データの組合せと、該順序値を有する画素におけるドット形成の有無との対応関係を記憶しておく第3の機能と、

前記個数データを生成した画素群内の各画素について、該個数データと前記順序値との組合せ毎に前記対応関係を参照することにより、ドット形成の有無を決定する第4の機能と、

前記決定したドット形成の有無に従って前記出力媒体上にドットを形成する第5の機能と

を実現することを要旨とする。

【0033】

また、上記のプログラムに対応する本発明の第1の記録媒体は、

画像データに所定の画像処理を施して、得られた結果に基づいて出力媒体上にドットを形成することによって画像を出力するプログラムを、コンピュータで読み取り可能に記録した記録媒体であって、

互いに隣接する画素を所定個数ずつ画素群にまとめて前記画像を複数の画素群に分割し、該各々の画素群内に形成されるドットの個数を表す個数データを、前記画像データに基づいて生成する第1の機能と、

前記画素群内の各画素について、該画素群内でドットが形成される順番を示す順序値を取得する第2の機能と、

前記順序値および前記個数データの組合せと、該順序値を有する画素におけるドット形成の有無との対応関係を記憶しておく第3の機能と、

前記個数データを生成した画素群内の各画素について、該個数データと前記順序値との組合せ毎に前記対応関係を参照することにより、ドット形成の有無を決定する第4の機能と、

前記決定したドット形成の有無に従って前記出力媒体上にドットを形成する第5の機能と

をコンピュータを用いて実現するプログラムを記録していることを要旨とする。

【0034】

更に、上述した画像出力方法に対応する本発明の第2のプログラムは、

所定の画像処理が施された画像データを受け取って、該画像データに基づいて出力媒体上にドットを形成することにより画像を出力する方法を、コンピュータを用いて実現するためのプログラムであって、

互いに隣接する画素が所定個数ずつ画素群としてまとめられた状態で、該画素群内に形成すべきドットの個数を表す個数データを、前記画像データとして受け取る機能（A）と

前記画素群内の各画素について、該画素群内でドットが形成される順番を示す順序値を取得する機能（B）と、

前記順序値および前記個数データの組合せと、該順序値を有する画素におけるドット形成の有無との対応関係を記憶しておく機能（C）と、

前記個数データを受け取った画素群内の各画素について、該個数データと前記順序値との組合せ毎に前記対応関係を参照することにより、ドット形成の有無を決定する機能（D）と、

前記決定したドット形成の有無に従って前記出力媒体上にドットを形成する機能（E）と

を実現することを要旨とする。

【0035】

また、上記のプログラムに対応する本発明の第2の記録媒体は、

所定の画像処理が施された画像データを受け取って、該画像データに基づいて出力媒体上にドットを形成することにより画像を出力するプログラムを、コンピュータで読み取り可能に記録した記録媒体であって、

互いに隣接する画素が所定個数ずつ画素群としてまとめられた状態で、該画素群内に形

成すべきドットの個数を表す個数データを、前記画像データとして受け取る機能（A）と、前記画素群内の各画素について、該画素群内でドットが形成される順番を示す順序値を取得する機能（B）と、前記順序値および前記個数データの組合せと、該順序値を有する画素におけるドット形成の有無との対応関係を記憶しておく機能（C）と、前記個数データを受け取った画素群内の各画素について、該個数データと前記順序値との組合せ毎に前記対応関係を参照することにより、ドット形成の有無を決定する機能（D）と、前記決定したドット形成の有無に従って前記出力媒体上にドットを形成する機能（E）とをコンピュータを用いて実現するプログラムを記録していることを要旨とする。

【0036】

こうしたプログラム、あるいは記録媒体に記録されているプログラムをコンピュータに読み込ませ、該コンピュータを用いて上述の各種機能を実現させれば、簡便に且つ迅速に画像を出力することが可能となる。

【発明を実施するための最良の形態】

【0037】

以下では、本発明の作用・効果をより明確に説明するために、本発明の実施の形態を、次のような順序に従って説明する。

A. 発明の概要：

B. 第1実施例：

B-1. 装置構成：

B-2. 第1実施例の画像印刷処理の概要：

B-3. 個数データに基づいて各画素のドット形成有無を決定可能な原理：

B-4. 第1実施例の個数データ生成処理：

B-5. 第1実施例のドット形成有無決定処理：

B-6. 第1実施例の変形例：

C. 第2実施例：

C-1. 第2実施例の画像印刷処理：

C-2. 第2実施例の個数データ生成処理：

C-3. 第2実施例のドット形成有無決定処理：

C-3-1. 変換テーブルを参照せずにドット形成有無を決定する方法：

C-3-2. 変換テーブルを参照するドット形成有無決定処理：

【0038】

A. 発明の概要：

実施例についての詳細な説明に入る前に、図1を参照しながら、本発明の概要について説明しておく。図1は、印刷システムを例にとって、本発明の概要を説明するための説明図である。本印刷システムは、画像処理装置としてのコンピュータ10と、画像出力装置としてのプリンタ20等から構成されており、コンピュータ10に所定のプログラムがロードされて実行されると、コンピュータ10およびプリンタ20などが全体として、一体の画像出力システムとして機能する。プリンタ20は、印刷媒体上にドットを形成することによって画像を印刷する。コンピュータ10は、印刷しようとする画像の画像データに所定の画像処理を施すことによって、プリンタ20が画素毎にドットの形成を制御するためのデータを生成して、該プリンタ20に供給する。

【0039】

一般的な印刷システムでは、次のようにして画像を印刷する。先ず、コンピュータで所定の画像処理を施すことにより、画像データを、画素毎にドット形成の有無を表すデータに変換する。次いで、得られたデータをプリンタに供給し、プリンタでは供給されたデータに従ってドットを形成することによって画像を印刷している。ここで、印刷しようとす

る画像の画素数が多くなると、それに伴って画像データのデータ量も大きくなるので、画像処理やデータの取り扱いのために要する時間が増加してしまい、画像を迅速に印刷することが困難となる。

【0040】

こうした点に鑑みて、図1に例示した印刷システムでは、次のようにして画像を印刷する。先ず、コンピュータ10では、画像を構成する画素を互いに隣接した所定個数ずつ画素群としてまとめることで、画像を複数の画素群に分割する。そして、各画素群について、画素群内に形成するドット個数を表す個数データを生成する。コンピュータ10に設けられた個数データ生成モジュールは、画像を複数の画素群に分割して、各画素群について個数データを生成する。こうして生成された個数データは、コンピュータ10に設けられた個数データ供給モジュールからプリンタ20に向かって供給される。

【0041】

プリンタ20に設けられたドット形成有無決定モジュールは、個数データ供給モジュールから供給された個数データを受け取ると、画素群内の各画素についてドット形成の有無を決定する。次いで、ドット形成モジュールが、各画素について決定されたドット形成の有無に従って、印刷媒体上にドットを形成することによって画像を印刷する。

【0042】

ここで、画素毎にドット形成の有無を表すデータに比べれば、画素群毎の個数データは遙かに小さなデータ量とすることができます。従って、コンピュータ10から画素毎にドット形成の有無を表したデータをプリンタ20に供給する代わりに、画素毎の個数データを供給してやれば、プリンタ20は極めて迅速にデータを受け取ることが可能である。

【0043】

また、プリンタ20は、個数データを受け取ると、次のようにして、画素群内に含まれる各画素についてのドット形成の有無を決定する。先ず、画素群内の各画素について、画素群内でドットが形成される順番を示す順序値を、順序値記憶モジュールに記憶しておく。また、順序値および個数データの組合せと、該順序値を有する画素についてのドット形成の有無との対応関係を、対応関係記憶モジュールに予め記憶しておく。そして、画素群についての個数データを受け取ると、画素群内の各画素についての順序値を取得して、個数データと順序値との組合せ毎に対応関係を参照することによって、各画素についてのドット形成の有無を決定する。このように対応関係を参照しながらドット形成の有無を決定してやれば、個数データを迅速に変換することが可能である。従って、コンピュータ10から個数データを迅速に受け取ることが可能であることと相まって、迅速に画像を印刷することが可能となる。以下では、こうした印刷システムを例にとって、本発明の各種実施例について詳細に説明する。

【0044】

B. 第1実施例：

B-1. 装置構成：

図2は、本実施例の画像処理装置としてのコンピュータ100の構成を示す説明図である。コンピュータ100は、CPU102を中心に、ROM104やRAM106などを、バス116で互いに接続して構成された周知のコンピュータである。

【0045】

コンピュータ100には、フレキシブルディスク124やコンパクトディスク126等のデータを読み込むためのディスクコントローラDDC109や、周辺機器とデータの授受を行うための周辺機器インターフェースP1F108、CRT114を駆動するためのビデオインターフェースV1F112等が接続されている。P1F108には、後述するカラープリンタ200や、ハードディスク118等が接続されている。また、デジタルカメラ120やカラースキャナ122等をP1F108に接続すれば、デジタルカメラ120やカラースキャナ122で取り込んだ画像を印刷することも可能である。また、ネットワークインターフェースカードNIC110を装着すれば、コンピュータ100を通信回線300に接続して、通信回線に接続された記憶装置310に記憶されているデータを取

得することもできる。

【0046】

図3は、本実施例のカラープリンタ200の概略構成を示す説明図である。カラープリンタ200はシアン、マゼンタ、イエロ、ブラックの4色インクのドットを形成可能なインクジェットプリンタである。もちろん、これら4色のインクに加えて、染料または顔料濃度の低いシアン（淡シアン）インクと、染料または顔料濃度の低いマゼンタ（淡マゼンタ）インクとを含めた合計6色のインクドットを形成可能なインクジェットプリンタを用いることもできる。尚、以下では場合によって、シアンインク、マゼンタインク、イエロインク、ブラックインク、淡シアンインク、淡マゼンタインクのそれぞれを、Cインク、Mインク、Yインク、Kインク、LCインク、LMインクと略称することがあるものとする。

【0047】

カラープリンタ200は、図示するように、キャリッジ240に搭載された印字ヘッド241を駆動してインクの吐出およびドット形成を行う機構と、このキャリッジ240をキャリッジモータ230によってプラテン236の軸方向に往復動させる機構と、紙送りモータ235によって印刷用紙Pを搬送する機構と、ドットの形成やキャリッジ240の移動および印刷用紙の搬送を制御する制御回路260などから構成されている。

【0048】

キャリッジ240には、Kインクを収納するインクカートリッジ242と、Cインク、Mインク、Yインクの各種インクを収納するインクカートリッジ243とが装着されている。インクカートリッジ242、243をキャリッジ240に装着すると、カートリッジ内の各インクは図示しない導入管を通じて、印字ヘッド241の下面に設けられた各色毎のインク吐出用ヘッド244ないし247に供給される。

【0049】

図4は、インク吐出用ヘッド244ないし247におけるインクジェットノズルNzの配列を示す説明図である。図示するように、インク吐出用ヘッドの底面には、C、M、Y、Kの各色のインクを吐出する4組のノズル列が形成されており、1組のノズル列あたり48個のノズルNzが、一定のノズルピッチkで配列されている。

【0050】

制御回路260は、CPUや、ROM、RAM、PIF（周辺機器インターフェース）等がバスで相互に接続されて構成されている。制御回路260は、キャリッジモータ230および紙送りモータ235の動作を制御することによってキャリッジ240の主走査動作および副走査動作を制御するとともに、コンピュータ100から供給されるデータに基づいて、各ノズルから適切なタイミングでインク滴を吐出する制御を行う。こうして、制御回路260の制御の下、印刷媒体上の適切な位置に各色のインクドットを形成することによって、カラープリンタ200はカラー画像を印刷することができる。

【0051】

また、インク滴を吐出するためにノズルに供給される駆動信号波形を制御してやれば、吐出されるインク滴の大きさを変更して、大きさの異なるインクドットを形成することもできる。このようにしてインクドットの大きさを制御すれば、印刷しようとする画像の領域に応じて異なる大きさのインクドットを使い分けて形成することで、より高画質の画像を印刷することも可能となる。

【0052】

尚、各色のインク吐出ヘッドからインク滴を吐出する方法には、種々の方法を適用することができる。すなわち、ピエゾ素子を用いてインクを吐出する方式や、インク通路に配置したヒータでインク通路内に泡（バブル）を発生させてインク滴を吐出する方法などを用いることができる。また、インクを吐出する代わりに、熱転写などの現象を利用して印刷用紙上にインクドットを形成する方式や、静電気を利用して各色のトナー粉を印刷媒体上に付着させる方式のプリンタを使用することも可能である。

【0053】

以上のようなハードウェア構成を有するカラープリンタ200は、キャリッジモータ230を駆動することによって、各色のインク吐出用ヘッド244ないし247を印刷用紙Pに対して主走査方向に移動させ、また紙送りモータ235を駆動することによって、印刷用紙Pを副走査方向に移動させる。制御回路260は、キャリッジ240の主走査および副走査の動きに同期させながら、適切なタイミングでノズルを駆動してインク滴を吐出することによって、カラープリンタ200は印刷用紙上にカラー画像を印刷している。

【0054】

B-2. 第1実施例の画像印刷処理の概要：

以下では、上記のようなコンピュータ100およびカラープリンタ200が、画像を印刷するために、それぞれの内部で行われる画像処理（画像印刷処理）について説明する。ここでは、理解の便宜を図るために、初めに画像印刷処理の全体像について簡単に説明し、次に、こうした画像印刷処理が可能である原理について説明する。そして最後に、それぞれの処理の詳細な内容について説明する。

【0055】

尚、以下では、画像印刷処理の前半部分はコンピュータ100で実施され、後半部分はカラープリンタ200で実施されるものとして説明するが、コンピュータ100が行う処理をカラープリンタ200の内部で実施したり、あるいはデジタルカメラ120など、画像データを生成する機器の内部で実施することとしてもよい。

【0056】

図5は、第1実施例の画像印刷処理の全体的な流れを示すフローチャートである。以下では、図5を参照しながら、画像印刷処理の全体像について簡単に説明する。第1実施例の画像印刷処理を開始すると、先ず初めに、コンピュータ100が画像データの読み込みを開始する（ステップS100）。ここでは、画像データはRGBカラー画像データであるものとして説明するが、カラー画像データに限らず、モノクロ画像データについても同様に適用することができる。また、カラープリンタに限らず単色プリンタについても同様に適用することができる。

【0057】

カラー画像データの読み込みに続いて、色変換処理を行う（ステップS102）。色変換処理とは、R, G, Bの階調値の組合せによって表現されているRGBカラー画像データを、印刷のために使用されるインク各色についての階調値の組合せによって表現された画像データに変換する処理である。前述したように、カラープリンタ200はC, M, Y, Kの4色のインクを用いて画像を印刷している。そこで、第1実施例の色変換処理ではRGB各色によって表現された画像データを、C, M, Y, Kの各色の階調値によって表現されたデータに変換する。色変換処理は、色変換テーブル（LUT）と呼ばれる3次元の数表を参照することで行う。LUTには、RGBカラー画像データに対して、色変換によって得られるC, M, Y, K各色の階調値が予め記憶されている。ステップS102の処理では、このLUTを参照することにより、RGBカラー画像データをC, M, Y, K各色の画像データに迅速に色変換することが可能となっている。

【0058】

色変換処理を終了すると、解像度変換処理を開始する（ステップS104）。解像度変換処理とは、画像データの解像度を、プリンタ200が画像を印刷する解像度（印刷解像度）に変換する処理である。画像データの解像度が印刷解像度よりも低い場合は、補間演算を行って画素間に新たな画像データを生成し、逆に画像データの解像度が印刷解像度よりも高い場合には、一定の割合でデータを間引くことによって、画像データの解像度を印刷解像度に一致させる処理を行う。

【0059】

以上のようにして解像度を印刷解像度に変換したら、コンピュータ100は、個数データ生成処理を開始する（ステップS106）。個数データ生成処理の詳細な内容は後ほど詳しく説明することとして、ここでは概要のみを説明する。個数データ生成処理では、隣接する画素を所定個数ずつ画素群としてまとめることにより、1つ画像を複数の画素群に

分割する。そして、それぞれの画素群の中で形成すべきドットの個数を表すデータ、すなわち個数データを画素群毎に決定する。一般に、ある画素にドットが形成されるか否かは、その画素の画像データに依存して決まるから、画素群内に形成されるドットの個数を表す個数データについても、画素群についての画像データに基づいて決定することができる。次いで、各画素群について決定した個数データを、カラープリンタ200に向かって出力する。個数データ生成処理では、このようにして、各画素についての画像データに基づいて個数データを画素群毎に生成した後、カラープリンタ200に供給する処理を行う。

【0060】

カラープリンタ200の制御回路260に内蔵されたCPUは、コンピュータ100から供給された個数データを受け取ると、ドット形成有無決定処理を開始する（ステップS108）。詳細な処理内容については後述するが、ドット形成有無決定処理では大まかには次のような処理を行う。上述したように、コンピュータ100から供給される個数データは、画素群に形成すべきドットの個数を表すデータであり、その個数のドットを画素群内のいずれの画素に形成するかについては未確定な状態となっている。そこで、画像を印刷するに先立って、画素群内の各画素についてドット形成の有無を決定しておく必要がある。ドット形成有無決定処理では、画素群内の各画素について、該画素群内でドットが形成される順番を示す順序値を記憶しておく、該順序値および個数データに基づいてドット形成の有無を決定することで、迅速に決定することが可能となっている。ドット形成有無を決定する処理の詳細についても後述する。

【0061】

以上のようにして、画素群内の各画素についてドット形成の有無を決定したら、決定したドット形成の有無に従って、出力媒体上にドットを形成する処理を行う（ステップS110）。すなわち、図3を用いて説明したように、キャリッジ240の主走査および副走査を繰り返しながらインク吐出用ヘッドを駆動してインク滴を吐出することにより、印刷用紙上にインクのドットを形成する。こうしてドットを形成することにより、画像データに対応した画像が印刷されることになる。

【0062】

このように、第1実施例の画像印刷処理では、コンピュータ100からカラープリンタ200に向かって、画素群に形成すべきドット個数のデータのみを供給しており、画素群に含まれる各画素についてのドット形成の有無に関するデータまでは供給していない。画素毎にドット形成の有無を表すデータに比べれば、画素群に形成するドットの個数は遙かに少ないデータ量で表現することができることから、このような方法を採用することで、コンピュータ100からカラープリンタ200に向かって極めて迅速にデータを供給することが可能となる。

【0063】

例えば、1つの画素群に8つの画素がまとめられており、形成可能なドットは1種類であるとする。この場合、各画素はドットが形成されるか否かのいずれかの状態しか取り得ないから、画素あたり1ビットのデータ長があれば表現することができる。画素群には8つの画素が含まれているから、画素群内の全画素についてドット形成の有無を表現しようとすると、8ビットのデータ長が必要となる。一方、画素群内に形成されるドットの個数は、0個～8個のいずれかの9通りしか取り得ない。9通りであれば4ビットであれば表現することができるから、画素群に形成するドットの個数は4ビットのデータ長で表現可能となる。このように、画素毎にドット形成の有無を表すデータに比べて、画素群内に形成されるドット個数は遙かに少ないデータ量で表現することができる。

【0064】

このように、画素毎にドット形成の有無を表すデータに比べれば、画素群に形成するドットの個数は遙かに少ないデータ量で表現することができる。第1実施例の画像印刷処理では、コンピュータ100からカラープリンタ200に向かって、画素群に形成すべきドット個数のデータのみを供給することにより、コンピュータ100からカラープリンタ200に向かって極めて迅速にデータを供給することが可能となる。

【0065】

また、カラープリンタ200は、コンピュータ100から画素群毎に個数データを受けると、画素群内の各画素についてのドット形成の有無を、後述する方法を用いて決定することで、極めて迅速に決定することが可能となっている。加えて、詳細には後述するが、各画素についてのドット形成の有無を適切に決定してやれば、ドット個数のデータのみを供給した場合でも、画質が悪化することはない。特に、後述する所定の条件においては、画素毎にドット形成の有無を表すデータを供給した場合と全く同じ結果を得ることが可能である。

【0066】

B-3. 個数データに基づいて各画素のドット形成有無を決定可能な原理：

以下では、上述した方法を採用した場合、すなわち、コンピュータ100からは画素群に形成するドット個数のデータを供給し、このドット個数のデータに基づいて、各画素についてのドット形成の有無をカラープリンタ200側で決定した場合でも、画質を悪化させることなく画像を印刷することが可能な原理について説明する。

【0067】

説明の都合上、先ず初めに、ディザ法について説明する。ディザ法とは、画像データを画素毎にドット形成の有無を表すデータに変換するために用いられる代表的な手法である。この手法では、ディザマトリックスと呼ばれるマトリックスに閾値を設定しておき、画像データの階調値とディザマトリックスに設定されている閾値とを画素毎に比較して、画像データの階調値の方が大きい画素についてはドットを形成すると判断し、そうでない画素についてはドットを形成しないと判断する。このような判断を画像中の全画素について行えば、画像データを画素毎にドット形成の有無を表すデータに変換することができる。

【0068】

図6は、ディザマトリックスの一部を概念的に例示した説明図である。図示したマトリックスには、横方向（主走査方向）に128画素、縦方向（副走査方向）に64画素、合計8192個の画素に、階調値1～255の範囲から万遍なく選択された閾値がランダムに記憶されている。ここで、閾値の階調値が1～255の範囲から選択されているのは、本実施例では画像データが階調値0～255の値を取り得る1バイトデータであるとしていることに加えて、画像データの階調値と閾値とが等しい場合には、その画素にはドットを形成するものと判断していることによる。

【0069】

すなわち、ドットが形成されるのは画像データの階調値が閾値よりも大きい画素に限る（すなわち階調値と閾値が等しい画素にはドットは形成しない）とした場合、画像データの取り得る最大階調値と同じ値の閾値を有する画素には、決してドットが形成されることはない。こうしたこと为了避免するため、閾値の取り得る範囲は、画像データの取り得る範囲から最大階調値を除いた範囲とする。逆に、画像データの階調値と閾値が等しい画素にもドットを形成するとした場合、画像データの取り得る最小階調値と同じ値の閾値を有する画素には、常にドットが形成されてしまうことになる。こうしたこと为了避免するため、閾値の取り得る範囲は、画像データの取り得る範囲から最小階調値を除いた範囲とする。本実施例では、画像データの取り得る階調値が0～255であり、画像データと閾値が等しい画素にはドットを形成するとしていることから、閾値の取り得る範囲を1～255としておくのである。尚、ディザマトリックスの大きさは、図6に例示したような大きさに限られるものではなく、縦と横の画素数が同じマトリックスも含めて種々の大きさとすることができる。

【0070】

図7は、ディザマトリックスを参照しながら、各画素についてのドット形成の有無を判断している様子を概念的に示した説明図である。ドット形成の有無を判断するに際しては、先ず、判断しようとする画素を選択し、この画素についての画像データの階調値と、ディザマトリックス中で対応する位置に記憶されている閾値と比較する。図7中に示した細い破線の矢印は、画像データの階調値と、ディザマトリックスに記憶されている閾値とを

、画素毎に比較していることを模式的に表したものである。例えば、画像データの左上隅の画素については、画像データの階調値は97であり、ディザマトリックスの閾値は1であるから、この画素にはドットを形成すると判断する。図7中に実線で示した矢印は、この画素にはドットを形成すると判断して、判断結果をメモリに書き込んでいる様子を概念的に表したものである。一方、この画素の右隣の画素については、画像データの階調値は97、ディザマトリックスの閾値は177であり、閾値の方が大きいので、この画素についてはドットを形成しないと判断する。ディザ法では、こうしてディザマトリックスを参照しながら、画素毎にドットを形成するか否かを判断することで、画像データを画素毎にドット形成の有無を表すデータに変換する。

【0071】

図8は、ディザ法を用いて画像データをドット形成の有無を表すデータに変換している様子を示した説明図である。図8（a）は、画像データの一部を拡大して示したものであり、図中の小さな矩形は画素を、そして、それぞれの矩形の中に表示された数値は画像データの階調値を表している。図示されているように、画像データは、隣接する画素間では近似する（若しくは同一の）階調値が割り当てられる傾向がある。こうした傾向は、高画質化の要請から画像データの解像度は高くなる傾向にあるが、隣接する画素間で近似若しくは同一の階調値が割り当てられる傾向は、画像データの解像度が高くなるほど顕著となっている。

【0072】

図8（b）は、ディザマトリックスの対応する位置に閾値が設定されている様子を示している。図8（a）に示した画像データの階調値と、図8（b）に示したディザマトリックスの閾値とを画素毎に比較することによって、ドット形成の有無を判断する。図8（c）は、こうして画素毎にドット形成の有無を判断した結果を示しており、図中で斜線を付した画素がドットを形成すると判断された画素である。

【0073】

ここで、隣接する画素を所定数ずつ画素群としてまとめ、画素群内でドットを形成すると判断された画素の個数を数えることを考える。一例として、主走査方向（図8中では横方向）に4画素分、副走査方向（図8中では縦方向）に2画素分の、合計8画素ずつを画素群としてまとめるものとする。図8（d）は、こうしてまとめられたそれぞれの画素群について、ドットを形成すると判断された画素を数えることによって得られたドット個数を示している。第1実施例の画像印刷処理において、コンピュータ100からカラープリンタ200に供給されるのは、このような画素群毎の個数のデータである。個数データには、ドットを形成する画素位置に関する情報は含まれていないが、次のようにすれば、個数データからドットを形成する画素位置の情報を復元して、画素毎にドット形成の有無を表すデータを生成することができる。

【0074】

図9は、個数のデータから、画素毎にドット形成の有無を表すデータを生成する様子を示した説明図である。図9（a）は、図8で画素群毎に形成するドットの個数を数えて得られた値を表している。また、図9（b）は、図8で画素毎にドット形成の有無を判断するために参照したディザマトリックスを示している。前述したようにディザ法では、画像データの階調値と、ディザマトリックスの対応する画素位置に設定された閾値とを比較して、画像データの階調値の方が大きければ、その画素にはドットを形成すると判断しており、ディザマトリックスの閾値が小さくなるほどドットが形成され易くなる。このことから、ディザマトリックスはドットが形成される画素の序列を表していると考えることができる。

【0075】

ディザマトリックスの有するこうした性質に着目すれば、画素群内に形成されるドットの個数から、ドットが形成される画素位置を決定することができる。例えば、図9（a）に示した一番左上隅の画素群について説明すると、この画素群に形成されるドット個数は3である。また、図9（b）に示したディザマトリックスを参照すれば、この画素群内で

は、左上隅にある画素位置すなわち閾値「1」が設定されている画素位置が最もドットが形成され易い画素であると言える。従って、この画素群内で3つ形成されるドットの中の1つは、左上隅の画素に形成されているものと考えることができる。同様にして、残りの2つのドットは、この画素群内で2番目にドットが形成され易い画素（すなわち図9（b）のディザマトリックスで閾値「42」が設定されている画素）と、3番目にドットが形成され易い画素（すなわち閾値「58」が設定されている画素）とに形成されるものと考えることができる。

【0076】

もちろん、ドット形成の有無は、ディザマトリックスに設定された閾値だけでなく、画像データの階調値によっても影響されるから、画像データの階調値が極端に大きければ、より小さな閾値が設定されている画素よりも先にドットが形成されることも起こり得る。しかし、前述したように画像データには、隣接する画素には近似する（若しくは同一の）階調値が割り当てる傾向があるから、ほとんどの場合はドットが形成され易い画素（すなわちディザマトリックスに設定された閾値の小さな画素）からドットが形成されると考えることができる。

【0077】

図9（a）に示した他の画素群についても、同様にして、ドット個数とディザマトリックスの閾値とに基づいて、ドットを形成する画素位置を決定することができる。例えば、図9（a）の上述した画素群の下にある画素群（左端上から2番目の画素群）については、ドット個数は3個であるから、図9（b）のディザマトリックスを参照すれば、これら3つのドットは、閾値「22」が設定された画素と、閾値「33」が設定された画素と、閾値「91」が設定された画素とに、それぞれ形成されると考えることができる。

【0078】

図9（a）に示した4つの画素群について、このようにして個数データからドットを形成する画素位置を決定すると、図9（c）に示した結果を得ることができる。図9（c）中で、斜線を付して示した画素はドットを形成すると判断された画素である。図9（c）と図8（c）とを比較すれば明らかのように、個数データから決定した画素位置は、画素毎に決定した画素位置と一致している。このことは、ディザマトリックスを参照して画素毎にドット形成の有無を判断し、画素群内に形成されるドットの個数のみを記憶しておけば、画素位置までは記憶していなくても、ディザマトリックスとドット個数とから、ドットが形成される画素位置を復元可能なことを示している。このことから、コンピュータ100から画素群毎の個数データを供給し、カラープリンタ200側で個数データからドットを形成する画素位置を決定した場合でも、画素位置を適切に決定して、画質を悪化させることなく画像を印刷することができる。

【0079】

また、個数データからドットを形成する画素位置を適切に決定するためには、画像データの階調値が画素群内で大きく異なっていなければ良い。前述したように、画像データは隣接する画素間では近似した階調値を有する特性があるから、こうした条件はほとんどの場合に成立しており、従って、個数データのみをカラープリンタ200に供給した場合でも、画質を悪化させることなく画像を印刷することができる。

【0080】

特に、次の2つの条件が満足される場合には、画像データの階調値とディザマトリックスの閾値とを比較してドット形成の有無を画素毎に判断した結果と、完全に同じ画素位置にドットを形成可能なことが保証される。先ず1つ目の条件は、画素群内で各画素の階調値が同一の値を有することであり、2つ目の条件は、コンピュータ100側で画素毎にドット形成の有無を判断する際に参照したディザマトリックスと、カラープリンタ200側で個数データから画素位置を決定するために参照するディザマトリックスとが、同一のマトリックスであることである。

【0081】

尚、図7で説明したディザ法においては、ディザマトリックスに設定された閾値と画像

データとの階調値とを比較して、いずれの値が大きいかによってドット形成の有無を判断している。これに対して、個数データから画素群内でドットが形成される画素位置を決定する場合には、図9を用いて説明したように、ドットが形成される画素位置を、ディザマトリックスに設定された閾値の小さな画素から順番に決定している。すなわち、画素位置を決定するためには、閾値の値まで必要なわけではなく、画素群内でドットが形成される順番が分かっていれば良い。このことから、図9 (b) に示すディザマトリックスの代わりに、図9 (d) に示すような画素群内の各画素について、ドットが形成される順番を示す値(順序値)が設定されたマトリックス(本明細書中では、このようなマトリックスを順序値マトリックスと呼ぶものとする)を記憶しておき、画素群毎に順序値マトリックスを参照しながら、個数データから画素位置を決定することも可能である。

【0082】

また、以上の説明では、図9に示したように、ディザマトリックスに基づいて生成された複数種類の順序値マトリックスを予め記憶しておき、画素群の個数データを受け取ると、その画素群に対応した順序値マトリックスを用いて、各画素についてのドット形成の有無を決定するものとして説明した。しかし、より簡便には次のようにしてドット形成の有無を決定しても良い。すなわち、予め複数の順序値マトリックスを記憶しておき、個数データを受け取ると、画素群毎にランダムに選択した1の順序値マトリックスを用いて、各画素についてのドット形成の有無を決定しても良い。更に、より簡便には、順序値マトリックスを1組だけ記憶しておき、このマトリックスを用いて各画素についてのドット形成の有無を決定することも可能である。

【0083】

B-4. 第1実施例の個数データ生成処理 :

以下では、図5に示した第1実施例の画像印刷処理において、画像データから個数データを生成する処理(図5のステップS106)について説明する。図10は、第1実施例の個数データ生成処理の流れを示すフローチャートである。以下、フローチャートに従つて、第1実施例の個数データ生成処理について説明する。

【0084】

第1実施例の個数データ生成処理を開始すると、先ず初めに、互いに隣接する所定個数の画素をまとめて画素群を生成する(ステップS200)。ここでは、主走査方向に4画素分、副走査方向に2画素分の合計8つの画素を画素群にまとめるものとする。尚、画素群としてまとめる画素は、このように矩形状に縦横の位置が揃った画素である必要はなく、互いに隣接し且つ所定の位置関係にあればどのような画素を画素群としてまとめて良い。

【0085】

次いで、画素群としてまとめた複数の画素の中から、処理対象として着目する画素(着目画素)を1つ設定する(ステップS202)。そして、着目画素に割り当てられた画像データの階調値とディザマトリックスの閾値とを比較することにより、着目画素についてのドット形成の有無を判断する(ステップS204)。すなわち、図7を用いて前述したように、画像データの方が大きい画素にはドットを形成するものと判断し、逆にディザマトリックスの閾値の方が大きい画素についてはドットを形成しないと判断する。

【0086】

次いで、画素群内の全ての画素について以上のようないしを用いたか否かを判断し(ステップS206)、画素群中に未処理の画素が残っている場合は(ステップS206:n0)、ステップS202に戻って続く一連の処理を行う。こうして画素群内の全ての画素について、ドット形成有無の判断を終了したら(ステップS206:y e s)、処理した画素群についての個数データを生成する(ステップS208)。ここでは、画素群内に形成されるドットの個数を計数し、得られたドットの個数を個数データとする。例えば、図8に示した画像データを処理するものとして説明すると、画像中の一番左上隅にある画素群については、画素群内に形成されるドットは3個であるから、個数データは「3」となる。

【0087】

以上のようにして、1つの画素群についての処理を終了したら、画像の全画素について処理を終了したか否かを判断し（ステップS210）、未処理の画素が残っていれば、ステップS200に戻って新たな画素群を生成した後、続く一連の処理を行って、その画素群の個数データを生成する（ステップS208）。こうした処理を繰り返して行くことにより、画像中の全画素についての処理を終了したら（ステップS210：yes）、各画素群について得られた個数データをカラープリンタ200に向かって出力して（ステップS212）、図10に示した個数データ生成処理を終了する。その結果、図8（d）に示すような、各画素群についての個数データがカラープリンタ200に供給されることになる。

【0088】

B-5. 第1実施例のドット形成有無決定処理：

次に、コンピュータ100から供給された個数データに基づいて、画素群内の各画素についてドット形成の有無を決定する処理（図5のステップS108）について説明する。図11は、第1実施例のドット形成有無決定処理の流れを示すフローチャートである。かかる処理は、カラープリンタ200の制御回路260に内蔵されたCPUによって実行される処理である。また、図12は、第1実施例のドット形成有無決定処理において、各画素についてのドット形成の有無が決定される様子を概念的に示した説明図である。以下では、図12を参照しながら、図11に示すフローチャートに従って、第1実施例のドット形成有無決定処理の内容について説明する。

【0089】

ドット形成有無決定処理を開始すると、先ず初めに、画素群を1つ選択し（ステップS300）、その画素群の個数データを取得する（ステップS302）。ここでは、図12（a）に示すような個数データが供給されたものとする。

【0090】

次いで、選択した画素群に含まれる各画素の中から対象とする画素を1つ選択し（ステップS304）、該画素群内で対象画素にドットが形成される順番を示す値（順序値）を取得する（ステップS306）。対象画素の順序値は、図12（b）に示すような、予め設定されている順序値マトリックスを参照することで容易に取得することができる。図12（b）に例示した順序値マトリックスには、画素群を構成する各画素の画素位置について、順序値が予め設定されている。例えば、画素群内で一番左上隅にある画素については、順序値「1」が設定されており、その画素の右隣の画素については、順序値「6」が設定されている。ステップS306では、このような順序値マトリックスを参照して、対象画素の位置に設定されている順序値を取得する。尚、説明が煩雑となることを避けるために、ここでは、順序値マトリックスは1組のみが記憶されており、常に同じ順序値マトリックスを用いて順序値を得るものとして説明するが、順序値マトリックスを複数組記憶しておき、画素群毎に順序値マトリックスを切り換ながら、対象画素の順序値を取得することとしても良い。

【0091】

こうして順序値を取得したら、変換テーブルを参照することによって、対象画素についてのドット形成の有無を決定する（ステップS308）。図13は、対象画素についてのドット形成の有無を決定するために参照される変換テーブルを概念的に示した説明図である。図示されているように、変換テーブルには、順序値と個数データとの組合せに対応付けて、ドット形成の有無が設定されている。ここでは、画素群は8つの画素から構成されているから、順序値は1～8までの値を取り、個数データは0～8までの値を取る。従って、変換テーブルには、これらを組合せた72通りの組合せに対応付けて、ドット形成の有無を示す値が設定されている。図13に示した例では、ドットを形成する組合せに「1」が、ドットを形成しない組合せには「0」が設定されている。

【0092】

一例として、図12に示した画素群内で一番左上隅にある画素について説明すると、図

12 (b) に示すように順序値は「1」であり、個数データは図12 (a) に示したように「3」である。図13の変換テーブルを参照すると、順序値「1」、個数データ「3」の組合せに対して設定されている値は「1」、すなわち、この画素についてはドットを形成するものと決定することができる。このように、図11のステップS308では変換テーブルを参照することで、画素群についての個数データと対象画素の順序値とから、対象画素についてのドット形成の有無を直ちに決定するのである。

【0093】

こうして、対象画素として選択した1つの画素についてドット形成の有無を決定したら、選択した画素群内の全画素について、ドット形成の有無を決定したか否かを判断する（ステップS310）。そして、画素群内で未だドット形成の有無を決定していない画素が残っている場合は（ステップS310：no）、ステップS304に戻って、画素群の中から新たな画素を対象画素として選択し、続く一連の処理を行う。

【0094】

こうした処理を繰り返し、画素群内の全画素についてドット形成の有無を決定したと判断されたら（ステップS310：yes）、今度は、個数データの供給された全画素群について処理を終了したか否かを判断する（ステップS312）。そして、未処理の画素群が残っていれば（ステップS312：no）、ステップS300に戻って新たな画素群を選択し、続く一連の処理を行う。こうした処理を繰り返すことにより、コンピュータから供給された個数データは、画素毎にドット形成の有無を示すデータに変換されていく。そして、全ての画素群について処理が終了したら（ステップS306：yes）、図11に示したドット形成有無決定処理を終了して、図5の画像印刷処理に復帰する。

【0095】

以上に説明した第1実施例のドット形成有無決定処理においては、画素群の個数データを受け取ると、順序値マトリックスを参照することによって対象画素の順序値を取得し、得られた順序値と個数データを用いて変換テーブルを参照するという極めて単純な方法によって、ドット形成の有無を決定することができる。従って、画素毎にドット形成の有無を表すデータを、個数データから極めて迅速に生成することができる。

【0096】

更に、ドット形成の有無を決定する処理は、記憶されているデータを参照しているだけなので、専用の論理回路を組み込んだチップを用いてハードウェア的に実行することも容易である。ドット形成の有無を決定する処理をハードウェア的に実行してやれば、より一層高速に処理することができる、それだけ画像を迅速に印刷することができる。

【0097】

加えて、近年のコンピュータでは処理の高速化を図るために、いわゆるパイプライン処理と呼ばれる技術が活用されているが、その一方で、条件分岐を含む処理に対しては、パイプライン処理の技術を適用しても、さほど処理を高速化することができず、返って処理速度が低下する場合もあることが知られている。上述した第1実施例のドット形成有無決定処理は、記憶されているデータを参照する処理がほとんどであり、条件分岐が含まれていないので、こうした観点からも、高速処理に適した処理であると言えることができる。

【0098】

このように、上述した第1実施例のドット形成有無決定処理は、迅速な処理を可能とする種々の要素を備えているため、どのような条件においても、個数データを、画素毎にドット形成の有無を表すデータに迅速に変換することができ、延いては画像を迅速に印刷することができる。

【0099】

B-6. 第1実施例の変形例：

上述した第1実施例のドット形成有無決定処理においては、予め1組または複数組の順序値マトリックスを設定しておき、常に同じ順序値マトリックスを参照して、あるいは参照する順序値マトリックスをランダムに切り換えるながら、各画素についてのドット形成の有無を決定するものとして説明した。しかし、図8および図9を用いて前述したように、

順序値マトリックスをディザマトリックスに基づいて生成し、画素群の位置に応じて適切な順序値マトリックスを参照しながらドット形成の有無を決定してやれば、より適切にドット形成の有無を決定することができ、従って高画質な画像を印刷することが可能となる。以下では、こうした第1実施例における変形例のドット形成有無決定処理について説明する。

【0100】

図14は、変形例のドット形成有無決定処理の流れを示すフローチャートである。以下、フローチャートに従って説明する。変形例のドット形成有無決定処理においても、先ず初めに画素群を1つ選択し（ステップS400）、その画素群の個数データを取得する（ステップS402）。次いで、複数組記憶されている順序値マトリックスの中から、選択した画素群に対応する順序値マトリックスを読み込む（ステップS404）。かかる処理について、図15および図16を参照しながら詳しく説明する。

【0101】

図15は、変形例のドット形成有無決定処理において参照される複数の順序値マトリックスを生成する方法について示した説明図である。前述したように、1つの画素群は、主走査方向には4画素ずつ、副走査方向には2画素ずつの合計8つの画素から構成されているとしているから、これと対応して、ディザマトリックスの閾値も、主走査方向に4画素分ずつ、副走査方向に2画素分ずつの合計8画素分ずつの閾値をブロックにまとめてやる。図15（a）は、ディザマトリックスの左上隅にある8画素分の閾値をブロックにまとめている様子を概念的に示した説明図である。ここではディザマトリックスは、図6に示したように、主走査方向には128画素分、副走査方向には64画素分の大きさを有するものとしているから、主走査方向に4画素分、副走査方向に2画素分ずつの画素をブロックにまとめれば、ディザマトリックスは主走査方向および副走査方向にそれぞれ32ブロックずつ、全体では1024個のブロックに分割されることになる。

【0102】

これらのブロックに、図15（b）に示すように、1番から1024番までの通し番号を付しておく。そして、1番から1024番までの各ブロックから1組ずつ順序値マトリックスを生成する。図15（c）は、通し番号1番のブロックから、順序値マトリックスを生成している様子を示した説明図である。図15（c）の左側半分には、通し番号1番のブロックに含まれるディザマトリックスの閾値が示されている。図7を用いて前述したように、ディザ法では画像データの階調値とディザマトリックスの閾値とを比較して、画像データの方が大きい場合にドットを形成すると判断しているから、ディザマトリックスの閾値が小さい画素ほどドットが形成され易くなる。従って、図15（c）に示した1番のブロックの中で1番最初にドットが形成される画素は、閾値「1」が設定された画素と考えることができる。そこで、この画素には順序値として「1」を設定する。同様に、2番目にドットが形成される画素は、2番目に小さな閾値である閾値「42」が設定された画素と考えることができる。そこで、この画素には順序値「2」を設定する。このようにして、ブロック内に設定されている閾値の小さな画素から順番に、順序値「1」から順序値「8」までを決定してやれば、図15（c）の右側半分に示した通し番号1番の順序値マトリックスを得ることができる。

【0103】

図15（d）は、同様にして、ブロック内で小さな閾値が設定されている画素から順番に、順序値「1」から順序値「8」までを設定することで、通し番号2番の順序値マトリックスを生成した様子を示している。図15（b）に示した通し番号「1」番から通し番号「1024」番までの全てのブロックについて、以上のような操作を行うことにより、通し番号「1」番から「1024」番までの順序値マトリックスを生成して記憶しておく。

【0104】

図14のステップS404では、これら「1」番から「1024」番の順序値マトリックスの中から、ドット形成の有無を決定しようとする画素群に対応するマトリックスを選

択して読み込む処理を行う。図16は、画素群に対応する順序値マトリックスを選択する方法を示した説明図である。今、ドット形成の有無を決定しようとしている画素群が、図16(a)に示すように、画像の一番左上隅を基準として主走査方向にn個目の画素群、副走査方向にm個目の画素群の位置にあるとする。また、このような画素群の位置を、座標値(n, m)によって表すものとする。

【0105】

ここで、ディザマトリックスの大きさは、通常は、画像のようにには大きくはない。このためディザ法では、1つのディザマトリックスを、画像データに対して少しずつ位置をずらしながら繰り返して使用している。これと同じ理由から、図14に示したドット形成有無決定処理においても、1つのディザマトリックスを少しずつ移動させながら繰り返して使用する。ディザマトリックスを移動させる方法は、ディザ法においては種々の方法が使用されており、ドット形成有無を決定する場合も種々の移動方法を適用することができるが、ここでは説明の便宜から、最も単純な方法すなわちディザマトリックスを主走査方向に移動させるものとして説明する。図16(b)には、ディザマトリックスを主走査方向に少しずつ移動させながら、繰り返して使用する様子が概念的に示されている。

【0106】

図15(a)に示したように、ディザマトリックスを分割するブロックの大きさは、個数データを生成した画素群の大きさと一致しているから、図16(b)に示すようにディザマトリックスを移動させていくと、ディザマトリックスの各ブロックは画素群の位置に一致する。換言すれば、全ての画素群には、ディザマトリックスを分割するいずれかのブロックが適用されることになる。

【0107】

今、処理しようとしている画素群には、ディザマトリックス中で主走査方向にN個目、副走査方向にM個目のブロックが適用されたものとする。図15(b)に示すように、ここでは、1つのディザマトリックスには主走査方向・副走査方向にそれぞれ32個ずつのブロックが含まれるとしており、また、処理しようとする画素群の座標値は(n, m)、すなわち、画像の左上隅を基準として主走査方向にn個目、副走査方向にm個目の位置にあるとしているから、N, Mはそれぞれ次式で求めることができる。

$$N = n - \text{int}(n/32) \times 32$$

$$M = m - \text{int}(m/32) \times 32$$

ここで、intは、小数点以下を切り捨てて整数化することを表す演算子である。すなわち、int(n/32)は、n/32の計算結果に対して小数点以下の数値を切り捨てるによって得られた整数値を表している。従って、ある画素群についてドット形成の有無を決定する場合には、画素群の座標値(n, m)から上式によってN, Mを求めた後、ディザマトリックス中で対応する位置にあるブロックの通し番号を取得して、そのブロックから生成された順序値マトリックスを用いればよい。

【0108】

もっとも実際には、M, Nの値は、図16(d)に示すような計算を実行せざとも、極めて簡便に求めることができる。以下、この点について説明する。図17は、画素群の座標値(n, m)から、適用する順序値マトリックスを選択する方法を具体的に示した説明図である。図17(a)は、数値nを表す10ビットの2進数データを概念的に示している。図17(a)では、各ビットを識別するために、最上位ビットから最下位ビットに向かって1番から10番までの通し番号を付して表示している。

【0109】

順序値マトリックスの選択に際しては、先ず初めに、int(n/32)を算出する。すなわち、数値nを32で除算して、小数点以下の値を切り捨てる操作を行う。32による除算は、2進数データを右方向に5ビット分だけビットシフトさせることで実行可能であり、また、データを整数形式で扱っていれば、小数点以下の値は自動的に切り捨てられてしまう。結局、int(n/32)の2進数データは、図17(a)に示した数値nの2進数データを、単に右方向に5ビット分だけビットシフトさせることで得ることができ

る。図17 (b) は、数値nをビットシフトして得られたint (n/32) の2進数データを概念的に表している。

【0110】

こうして得られたint (n/32) に32を乗算する。32による乗算は、2進数データを5ビット分だけ左方向にビットシフトすることで実施することができる。図17 (c) は、数値nをビットシフトして得られたint (n/32) × 32の2進数データを概念的に表している。

【0111】

次いで、数値nから、int (n/32) × 32を減算すれば、前述の数値Nを得ることができる。数値nの2進数データ（図17 (a) 参照）とint (n/32) × 32の2進数データ（図17 (c) 参照）とを比較すれば明らかのように、これら2進数データは、上位の5ビットは共通しており、減算する側の数値の下位5ビットは全て「0」となっている。従って、減算される側の数値（数値n）の下位5ビットをそのまま抜き出せば、求める数値Nを得ることができる。すなわち、図17 (a) に示した2進数データと、図17 (e) に示すようなマスクデータとの論理積を求めるだけで、極めて簡単に数値Nを得ることが可能である。

【0112】

図17では、画素群の座標値 (n, m) の数値nから、ディザマトリックス中のブロック位置を示す数値Nを求める場合について説明したが、全く同様にして、ブロック位置を示す数値Mについても、数値mから極めて簡便に求めることができる。結局、画素群の座標値 (n, m) が与えられれば、数値n, mから数値N, Mを求めるこことにより、その画素群には通し番号が何番の順序値マトリックスが適用されるかを知ることができる。図14に示したドット形成有無決定処理のステップS404では、このようにして、画素群に対応するマトリックスを選択して読み込む処理を行う。

【0113】

以上のようにして、画素群に対応する順序値マトリックスを読み込んだら、処理中の画素群の中から、ドット形成の有無を決定しようとする対象画素を1つ選択する（ステップS406）。そして、読み込んでおいた順序値マトリックスを参照することで、対象画素の順序値を取得した後（ステップS408）、変換テーブルを参照することによって、対象画素についてのドット形成の有無を決定する（ステップS410）。変換テーブルには、図13を用いて前述した第1実施例の変換テーブルと同様に、順序値と個数データとの組合せに対応付けて、ドット形成の有無が設定されている。従って、画素群の個数データと対象画素の順序値とが分かれば、変換テーブルを参照することで直ちにドット形成の有無を決定することができる。

【0114】

こうして、1つの対象画素についてドット形成の有無を決定したら、選択した画素群内の全画素について、ドット形成の有無を決定したか否かを判断する（ステップS412）。未だ、ドット形成の有無を決定していない画素が残っていれば（ステップS412：no）、ステップS406に戻って、新たな対象画素を選択し、続く一連の処理を行う。こうした処理を繰り返し、画素群内の全画素についてドット形成の有無を決定したと判断されたら（ステップS412：yes）、今度は、個数データが供給された全画素群について処理を終了したか否かを判断する（ステップS414）。そして、未処理の画素群が残っていれば（ステップS414：no）、ステップS400に戻って新たな画素群を選択し、続く一連の処理を行う。こうした処理を繰り返し、全ての画素群について処理が終了したら（ステップS414：yes）、図14に示した変形例のドット形成有無決定処理を終了して、図5の画像印刷処理に復帰する。

【0115】

以上に説明した変形例のドット形成有無決定処理においては、ディザマトリックスに基づいて複数の順序値マトリックスを生成する。そして、ある画素群についてドット形成の有無を決定する際には、ディザ法を適用したときに、その画素群の位置に適用される部分

のディザマトリックスから生成された順序値マトリックスを用いて、ドット形成の有無を決定する。こうすれば、ディザマトリックスを用いて得られたドットの分布に準じた分布が得られるように、ドット形成の有無を決定することができる。周知のように、ディザマトリックスには、ドットが適切な分布で形成されるように、適切な分布で閾値が設定されているから、ディザマトリックスによるドットの分布に準じた分布が得られれば、高画質な画像を印刷することが可能となる。

【0116】

更に、順序値マトリックスを生成するために用いるディザマトリックスと、図10に示した個数データ生成処理中で使用するディザマトリックスとを同じマトリックスとしておけば、図8および図9を用いて前述したように、個数データから復元したドット分布は、ほとんどの場合は、ディザ法を用いて画素毎にドット形成の有無を判断した場合と全く同じドット分布となる。もちろん、前述したように、画素群内で画像データの階調値が大きく変化している場合には、ドット分布は異なったものとなるが、画像データには隣接する画素間では近似した（若しくは同一の）階調値を有する傾向があるため、多くの場合、ドット分布は同一となる。従って、適切なドット分布となるようにドット形成の有無を決定することができ、それだけ高画質な画像を印刷することが可能となる。

【0117】

もっとも、上述した変形例のドット形成有無決定処理では、変換テーブルに加えて、複数（上述した例では1024個）の順序値マトリックスを記憶しておく必要がある。これらテーブルおよびマトリックスを記憶するために、あまりに多くのメモリを使用したのでは、実際の製品に搭載する上で好ましいことではない。しかし、以下に説明するように、変換テーブルおよび順序値マトリックスを記憶するために多くのメモリを使用することはない。

【0118】

先ず、変換テーブルを記憶するために必要なメモリ容量について説明する。図13に示したように、変換テーブルには、順序値および個数データの組合せ毎にドット形成の有無が設定されているから、変換テーブルのデータサイズは、順序値および個数データがそれぞれ取り得る個数と、1画素分のドット形成の有無を表すために必要なデータ長とによって決定される。順序値は、画素群内の各画素にドットが形成される順番を示しているから、順序値は1つの画素群に含まれる画素数と同じ種類の値を取り得る。また、個数データは、画素群内に形成され得るドットの個数を表しているから、0個から画素数までの値を取ることができるので、画素数+1通りの値を取り得る。更に、ここでは、1つの画素にはドットが形成されるか否かのいずれかの状態しか取らないとしているから、1画素分のドット形成の有無は、1ビットあれば表現することができる。結局、変換テーブルを記憶するためのメモリ容量は、画素群に含まれる画素数をnとすると、

$$n \times (n + 1) \text{ ビット}$$

あればよい。画素群に含まれる画素数は、高々16個程度だから、変換テーブルを記憶するためには要するメモリ量は僅かなものである。

【0119】

次に、順序値マトリックスを記憶するために必要なメモリ量について説明する。順序値マトリックスを記憶するためのメモリ量は、マトリックス1つ当たりに要するメモリ量と、マトリックスの個数とによって決定される。順序値マトリックスには、画素群内の各画素にドットが形成される順番が設定されているから、マトリックス1つ当たりのメモリ量は、画素群に含まれる画素数によって決まる。また、順序値マトリックスの個数は、図15を用いて前述したように、ディザマトリックスを画素群と同じ大きさのブロックで分割した時に得られるブロック数に等しいから、ディザマトリックスの大きさと、画素群の大きさとによって決定される。結局、順序値マトリックスを記憶するために必要なメモリ量は、ディザマトリックスの大きさと、画素群の大きさとによって決定されることになる。

【0120】

図18は、各種大きさのディザマトリックスと、各種大きさの画素群とを想定して、順

序値マトリックスを記憶するために必要なメモリ量を試算した結果を示す説明図である。具体的には、ディザマトリックスの大きさとしては、 64×64 （すなわち主走査方向に64画素、副走査方向に64画素）、 128×64 （主走査方向に128画素、副走査方向に64画素）、 128×128 （主走査方向に128画素、副走査方向に128画素）の3種類のサイズを想定している。画素群の大きさとしては、 2×2 （主走査方向に2画素、副走査方向に2画素）、 4×2 （主走査方向に4画素、副走査方向に2画素）、 4×4 （主走査方向に4画素、副走査方向に4画素）の3種類のサイズを想定している。上述した実施例に対応する条件、すなわち、ディザマトリックスサイズが 128×64 であり、画素群の大きさが 4×2 の条件での試算結果は、図18では、破線で囲って示されている。以下では、この条件を代表例として用いながら、順序値マトリックスの試算例について説明する。

【0121】

順序値マトリックスの個数は、ディザマトリックスを画素群と同じサイズで分割して得られるブロックの個数であるから、ディザマトリックスの画素数（ 128×64 ）を画素群当たりの画素数（ 4×2 ）で除算して、1024となる。また、順序値マトリックスに設定される順序値は、1～8までの8通りの値を取るから、1つの順序値は3ビットあれば表現することができる。順序値マトリックスには8つの順序値が設定されているから、1つの順序値マトリックスを記憶するために必要なメモリ量は、 $3 \times 8 = 24$ ビット（3バイト）となる。順序値マトリックスの個数は1024個あるから、全ての順序値マトリックスを記憶するために必要なメモリ量は、3Kバイトと求めることができる。

【0122】

また、画素群に含まれる画素数が4つの場合は、順序値は1～4までの4通りの値を取るから、1つの順序値は2ビットあれば表現することができる。順序値マトリックスには4つの順序値が設定されるから、1つの順序値マトリックスを記憶するために要するメモリ量は、 $2 \times 4 = 8$ ビット（1バイト）となる。同様に、画素群に含まれる画素数が16個の場合は、1つの順序値を表現するために要するデータ長は4ビット、これが16個あるから、順序値マトリックスを記憶するために要するメモリ量は、 $4 \times 16 = 64$ ビット（8バイト）となる。

【0123】

図18には、各種条件で、全ての順序値マトリックスを記憶するために必要なメモリ量を試算した結果がまとめて示されている。図示されている試算結果を見れば明らかのように、順序値マトリックスを記憶しておくために必要なメモリ量は、高々10Kバイトあれば十分と考えられる。このため、変換テーブルや順序値マトリックスを記憶するために必要なメモリ量が、実際の製品に搭載する上で障害になるほど大きくなることはない。

【0124】

C. 第2実施例：

以上に説明した第1実施例では、カラープリンタ200で形成可能なドットは1種類であるものとして説明した。しかし、今日では、印刷画質を向上させることを目的として、大きさの異なるドットや、インク濃度の異なるドットなど、多種のドットを形成可能なプリンタ（いわゆる多値ドットプリンタ）が広く使用されている。本願の発明は、こうした多値ドットプリンタに適用した場合にも、大きな効果を得ることができる。以下では、第2実施例として、本願発明を多値ドットプリンタに適用した場合について説明する。

【0125】

C-1. 第2実施例の画像印刷処理の概要：

第2実施例の画像印刷処理は、フローチャートについては、図5に示した第1実施例の画像印刷処理と同様である。以下では、図5のフローチャートを流用しながら、第2実施例の画像印刷処理の概要について簡単に説明する。

【0126】

第2実施例の画像印刷処理を開始すると、先ず初めに、コンピュータ100で画像データを読み込んだ後、色変換処理を行う（図5のステップS100およびステップS102

相当）。次いで、解像度変換処理を行って、画像データの解像度を印刷解像度に変換した後（ステップS104相当）、個数データ生成処理を開始する（ステップS106相当）。

【0127】

前述したように、第1実施例では、カラープリンタ200が形成可能なドットは1種類であるものとしており、個数データ生成処理では、画素群内に形成されるドット個数を表す個数データを生成して、カラープリンタ200に出力した。これに対して、第2実施例では、カラープリンタ200は、複数種類のドットを形成可能である。ここでは、大きさの異なる3種類のドット、すなわち大ドット、中ドット、小ドットを形成可能であるものとする。このことと対応して、第2実施例の個数データ生成処理では、画素群内に、大ドット、中ドット、小ドットがそれぞれ何個ずつ形成されるかを表すデータを、個数データを生成することになる。

【0128】

また、詳細には後述するが、個数データを少ないデータ量で効率よく出力するために、大ドット、中ドット、小ドットの個数をそのまま出力するのではなく、コード化された状態で出力する。こうした第2実施例の個数データ生成処理の詳細については後述する。尚、ここでは、カラープリンタ200が形成可能なドットは、大ドット、中ドット、小ドットと、互いに大きさの異なるドットであるものとして説明するが、もちろん、ドットの種類が異なるのであれば、大きさが異なる場合に限られるものではない。例えば、ドットを形成するインクの濃度が異なる複数種類のドットとしたり、あるいは、微細なドットを複数形成することで擬似的に1つのドットを形成する場合には、微細なドットの密度が異なる複数種類のドットとすることも可能である。

【0129】

カラープリンタ200の制御回路260に内蔵されたCPUは、コンピュータ100から供給された個数データを受け取ると、ドット形成有無決定処理を開始する（図5のステップS108相当）。詳細には後述するが、第2実施例のドット形成有無決定処理では、コード化された状態の個数データを受け取ると、画素群内の各画素について、大ドット、中ドット、小ドットのいずれのドットを形成するか、あるいはドットを形成しないかを決定する処理を行う。

【0130】

こうして、大中小の各種ドットについてドット形成の有無を決定したら、得られた結果に従って各種ドットを形成する（図5のステップS110相当）。こうして大ドット、中ドット、小ドットを形成することにより、画像データに対応した画像が印刷される。

【0131】

C-2. 第2実施例の個数データ生成処理 :

次に、上述した第2実施例の画像印刷処理において、画素群内に形成される大ドット、中ドット、小ドットの個数がコード化された個数データを生成する処理について説明する。

【0132】

図19は、画素群内に形成される大ドット、中ドット、小ドットの個数を決定して、個数データを生成する処理の流れを示すフローチャートである。尚、かかる処理の詳細については、特許3292104号に開示されている。以下、フローチャートに従って説明する。処理を開始すると先ず初めに、互いに隣接する所定数の画素をまとめて画素群を形成する（ステップS500）。ここでは、前述の第1実施例と同様に、主走査方向に4画素、副走査方向に2画素の合計8つの画素を画素群としてまとめるものとする。

【0133】

次いで、画素群の中からドット形成の有無を判断するべく、処理対象とする画素を1つ選択した後（ステップS502）、選択した処理画素について、大ドット、中ドット、小ドットの形成有無を判断する（ステップS504）。大中小ドットの形成有無は次のようにして判断する。

【0134】

図20は、選択した1つの画素についてハーフトーン処理を行うことにより、大ドット、中ドット、小ドットの形成有無を判断する処理の流れを示すフローチャートである。大中小ドットのハーフトーン処理を開始すると、先ず初めに処理対象とする画素についての画像データを、大ドット、中ドット、小ドットの各ドットについての密度データに変換する（ステップS550）。ここで、密度データとは、ドットをどの程度の密度で形成するかを表すデータである。密度データは、大きな階調値となるほどドットが高い密度で形成されることを表している。例えば、密度データの階調値「255」は、ドットの形成密度が100%、すなわち全ての画素にドットが形成されることを表しており、密度データの階調値「0」は、ドットの形成密度が0%、すなわちいずれの画素にもドットが形成されないことを表している。こうした密度データへの変換は、ドット密度変換テーブルと呼ばれる数表を参照することによって行うことができる。

【0135】

図21は、画像データの階調値を大中小各ドットについての密度データに変換する際に参照されるドット密度変換テーブルを概念的に示した説明図である。図示されているように、ドット密度変換テーブルには、色変換によって得られた画像データの階調値に対して、小ドット・中ドット・大ドットの各ドットについての密度データが設定されている。画像データが階調値「0」近傍の領域では、中ドット・大ドットの密度データはいずれも階調値「0」に設定されている。小ドットの密度データは、画像データの階調値が大きくなるに連れて増加していくが、画像データがある階調値に達すると今度は逆に減少し始め、代わりに中ドットの密度データが増加し始める。画像データの階調値が更に増加して、ある階調値に達すると、小ドットの密度データが階調値「0」となり、中ドットの密度データが減少し始めて、代わりに大ドットの密度データが少しずつ増加していく。図20のステップS552では、このドット密度変換テーブルを参照しながら、画像データの階調値を、大ドットの密度データ、中ドットの密度データ、小ドットの密度データに変換する処理を行う。

【0136】

このようにして、処理対象とする画素について、大中小各ドットの密度データが得られたら、先ず初めに大ドットについての形成有無を判断する（図20のステップS554）。かかる判断は、大ドットの密度データと、処理対象としている画素の対応する位置に設定されているディザマトリックスの閾値とを比較することによって行う。そして、密度データの方が大きい場合は処理対象の画素には大ドットを形成するものと判断し（ステップS554：yes）、ハーフトーン処理を抜けて図19に示した個数データ生成処理に復帰する。

【0137】

逆に、大ドットの密度データよりも閾値の方が大きい場合は処理対象の画素には大ドットは形成されないと判断して（ステップS554：no）、今度は中ドットについて形成有無を判断する処理を開始する。中ドットの形成有無の判断には、大ドットの密度データと中ドットの密度データとを加算して、中ドット用の中間データを算出する（ステップS556）。そして、得られた中ドット用の中間データと、ディザマトリックスの閾値とを比較することにより、中ドットの形成有無を判断する（ステップS558）。そして、中ドット用の中間データの方が大きい場合は処理対象の画素には中ドットを形成するものと判断し（ステップS560：yes）、ハーフトーン処理を抜けて図19の個数データ生成処理に復帰する。

【0138】

逆に、中ドット用の中間データよりも閾値の方が大きい場合は処理対象の画素には中ドットも形成されないと判断して（ステップS560：no）、今度は小ドットについて形成有無を判断する処理を開始する。小ドットの形成有無の判断には、中ドット用の中間データと小ドットの密度データとを加算して、小ドット用の中間データを算出する（ステップS562）。そして、得られた小ドット用の中間データと、ディザマトリックスの閾値

とを比較することにより、小ドットの形成有無を判断する（ステップS564）。そして、小ドット用の中間データの方が大きい場合は処理対象の画素には小ドットを形成するものと判断し、逆に、小ドット用の中間データよりも閾値の方が大きい場合には、いずれのドットも形成されないものと判断する。

【0139】

以上のような処理を行えば、処理対象としている画素について、大ドット、中ドット、小ドットのいずれのドットを形成するか、若しくは、いずれのドットも形成しないかを判断することができる。こうして、対象画素についてドット形成の有無を判断したら、図20に示したハーフトーン処理を抜けて図19の個数データ生成処理に復帰する。

【0140】

上述した処理を行いながら大中小の各ドットの形成有無を判断する様子について、図22を参照しながら補足して説明する。図22は、画素群内の各画素について、ディザ法を適用しながら大中小各ドットの形成有無を判断している様子を概念的に示した説明図である。ここでは、説明が煩雑となることを避けるために、画素群内の全画素が同じ階調値を有しており、従って、大中小各ドットの密度データも同じ階調値を有しているものとする。図22(a)は、画素群内の各画素について得られた大中小ドットの密度データを示しており、いずれの画素も、大ドットの密度データが「2」、中ドットの密度データが「90」、小ドットの密度データが「32」であったものとする。

【0141】

図22(b)は、ディザマトリックス中で、画素群に対応する位置に記憶されている閾値を表している。大ドットの形成有無を判断する際には、大ドットの密度データと、これら閾値とを比較する。ここでは、いずれの画素についても、大ドットの密度データは「2」であるとしているから、大ドットを形成すると判断される画素は、閾値「1」が設定された画素だけである。図22(b)には、大ドットが形成されると判断された画素には、細かい斜線を付して表示している。その他の画素については、中ドットか小ドットのどちらかが形成されるか、若しくはいずれのドットも形成されないかのいずれかであると考えられる。そこで、中ドットの形成有無を判断する。

【0142】

中ドットの形成有無の判断に際しては、大ドットの密度データ「2」と中ドットの密度データ「90」とを加算して中ドット用の中間データを算出し、得られた中間データ「92」とディザマトリックスの閾値とを比較する。その結果、閾値「42」が設定された画素と、閾値「58」が設定された画素の2つの画素にのみ、中ドットが形成されるものと判断される。図22(c)には、中ドットが形成されると判断された画素には、少し細かい斜線を付して表示している。そして、大ドットも中ドットも形成されない画素については、小ドットが形成されるか、ドットが形成されないかのいずれかであると考えられる。そこで、中ドット用の中間データ「92」に小ドットの密度データ「32」を加算して、小ドット用の中間データを算出し、得られた中間データ「124」とディザマトリックスの閾値とを比較する。その結果、閾値「109」が設定された画素にのみ、小ドットが形成されるものと判断される。図22(d)には、小ドットが形成されると判断された画素には、粗い斜線を付して表示している。

【0143】

図19に示した個数データ生成処理のステップS502～S506では、以上のようにして画素群内の各画素について中間データを算出しながら、大中小の各ドットについての形成有無を判断していく。こうして、画素群内の全画素について判断を終了したら（ステップS506：yes）、画素群内に形成される大ドット、中ドット、小ドットの個数を取得する（ステップS508）。図22に例示した画素群については、大ドット1個、中ドット2個、小ドット1個となる。

【0144】

以上のようにして、画素群内に形成される各種ドットの個数が得られたら、各種ドットの個数の組合せ（図22に示した例では、大ドット1個、中ドット2個、小ドット1個と

いう組合せ) をコード化する処理を行う(ステップS510)。これは、次のような理由によるものである。例えばドットの種類が大中小の3種類である場合、ドットの種類毎に形成する個数を出力したのでは、1つの画素群について3回ずつドット個数を出力しなければならず、コンピュータ100からカラープリンタ200にデータを迅速に出力することで画像を迅速に印刷するという効果が滅殺されてしまう。そこで、各ドットの個数を個別に出力するのではなく、各ドットの個数の組合せを、組合せ毎に設定された個別のコードに変換しておくのである。

【0145】

大中小各ドットの組合せをコード化する処理は、ドット個数の組合せとコード化された個数データとを対応付けた状態で予め対応テーブルに記憶しておき、この対応テーブルを参照することで行う。図23は、画素群に形成される大中小各ドットの個数の組合せと、コード化された個数データとが対応付けた対応テーブルを示す説明図である。図23に例示の対応テーブルには、例えば大ドット、中ドット、小ドットの個数がいずれも0個である組合せには、コード化された個数データ「0」が対応付けられている。また、大ドットが0個、中ドットが0個、小ドットが1個の組合せには、コード課された個数データ「1」が対応付けられている。このように対応テーブルには、各ドットの個数の組合せ毎に、コード化された個数データが予め設定されている。

【0146】

ここで、大中小ドットの個数の組合せ数は、次のようなものとなる。画素群内の各画素には、大ドット、中ドット、小ドットのいずれのドットも形成され得るが、1つの画素に複数のドットが形成されることはないから、ドット個数の合計が画素群内の画素数(上述した実施例では8個)を越えることはない。従って、これら大中小ドットの個数の組合せは、「大ドットを形成する」、「中ドットを形成する」、「小ドットを形成する」、「ドットを形成しない」の4つの状態の中から重複を許して8回選択するときの組合せの数に等しくなるから、

$${}^4H_8 \quad (= {}^4 + {}^8 - {}^1 C_8)$$

によって求められ、165通りの組合せが存在することになる。ここで、 ${}_n H_r$ は、n種類の物の中から重複を許してr回選択したときに得られる組合せの数(重複組合せ数)を求める演算子である。また、 ${}_n C_r$ は、n種類の物の中から重複を許さずにr回選択したときに得られる組合せの数を求める演算子である。

【0147】

このように、大中小各ドットの個数の組合せが165通り存在していることから、コードデータも「0」～「164」の165通りあればよい。165通りであれば、8ビットのデータ長があれば表現することができる。結局、大ドットの個数、中ドットの個数、小ドットの個数と、3回出力する代わりに、8ビットのコード化された個数データを1回出力するだけで、画素群に形成する各種ドットの個数を出力することが可能となる。そこで、図19のステップS510では、ステップS508で画素群毎に得られた各種ドットについての個数の組合せを、図23に示すような対応テーブルを参照しながらコード化された個数データに変換しておくのである。

【0148】

こうして、画素群毎のドット個数の組合せをコード化した個数データが得られたら、画像の全画素について以上の処理を行ったか否かを判断する(ステップS512)。そして、未処理の画素が残っている場合は、ステップS500に戻って続く一連の処理を繰り返し、画像の全画素について処理を終了したと判断されたら、コード化された個数データを出力して(ステップS514)、図19に示した個数データ生成処理を終了する。

【0149】

C-3. 第2実施例のドット形成有無決定処理：

次に、上述した第2実施例の画像印刷処理において、画素群毎に生成されたコード化された状態の個数データを受け取り、画素群内の各画素についてドット形成の有無を決定する処理について説明する。前述した第1実施例のドット形成有無決定処理では、変換テー

ブルを参照することで、個数データと順序値とから直ちにドット形成の有無を決定したが、第2実施例のドット形成有無決定処理においても同様に、変換テーブルを参照することにより、コード化された個数データと順序値とから直ちに大中小ドットの形成有無を決定することが可能である。このような処理が可能な理由を明らかにするとともに、こうした処理を行うことで処理が高速化されることを示すために、以下では、先ず初めに、変換テーブルを用いずに個数データからドット形成の有無を決定する処理について説明する。そして、次に、変換テーブルを参照することで、個数データから大中小各種ドットの形成有無を迅速に決定可能な、第2実施例のドット形成有無決定処理について説明する。

【0150】

C-3-1. 変換テーブルを参照せずにドット形成有無を決定する方法：

図24は、変換テーブルを参照することなく、大中小各種ドットの形成有無を決定する処理の流れを示すフローチャートである。以下では、フローチャートに従って簡単に説明する。処理を開始すると、先ず初めに、画素位置を決定しようとする画素群を1つ選択し（ステップS600）、その画素群の個数データを取得する（ステップS602）。こうして取得された個数データは、コード化されたデータとなっている。そこで、個数データを復号して、大ドット、中ドット、小ドットについての個数を示すデータに変換する処理を行う（ステップS604）。個数データを復号する処理は、復号テーブルを参照することによって行う。図25は、コード化された個数データを復号するために参照される復号テーブルを概念的に示した説明図である。

【0151】

図示されるように、復号テーブルには、コード化された個数データに対応する大ドット、中ドット、小ドットのドット個数の組合せが設定されている。例えば、コード化された個数データが「1」である場合は、大ドットおよび中ドットの個数は0個で、小ドットの個数が1個であるドット個数の組合せに復号される。図24のステップS604では、このような復号テーブルを参照することにより、コード化された個数データを大中小各ドットの個数を表すデータに変換する。

【0152】

次いで、個数データを復号した画素群に対応する順序値マトリックスを読み込んだ後（ステップS606）、読み込んだ順序値マトリックスを参照しながら、画素群内の各画素について大ドット、中ドット、小ドットの形成有無を判断していく。ここで順序値マトリックスとは、図15を用いて前述したように、画素群内の各画素についてドットが形成される順番を設定したマトリックスである。図26は、順序値マトリックスを参照しながら、大中小の各種ドットについての形成有無を画素毎に決定している様子を概念的に示した説明図である。例えば、個数データを復号することにより、大ドットの個数が1個、中ドットの個数が2個、小ドットの個数が1個の組合せが得られたものとする。

【0153】

各画素についてのドット形成の有無を決定するに際しては、先ず初めに大ドットを形成する画素を決定する（図24のステップS608）。ここでは、大ドットの個数は1個であるとしているから、最もドットの形成されやすい画素、すなわち、順序値マトリックスで順序値が「1」に設定されている画素に大ドットが形成されるものと判断する。ここで、大ドットの個数がN個である場合は、順序値マトリックスで順序値が「1」から「N」までの値が設定されている画素に大ドットを形成するものと判断する。図26では、大ドットを形成する画素には、細かい斜線を付して表示している。

【0154】

大ドットを形成する画素を決定したら、大ドットが形成されなかった画素の中から、今度は、中ドットを形成する画素を決定する（図24のステップS610）。図26に示すように、中ドットの個数は2個であり、順序値「1」が設定されている画素位置には大ドットが形成されているから、中ドットは順序値「2」が設定された画素と順序値「3」が設定された画素とに形成される。図26では、中ドットが形成される画素位置には、少し粗い斜線を付して表示している。

【0155】

こうして中ドットを形成する画素を決定したら、大ドットも中ドットも形成されない画素の中から、小ドットを形成する画素を決定する（図24のステップS612）。図26に示すように、小ドットの個数は1個であり、順序値「1」の画素には大ドットが、順序値「2」および「3」の画素には中ドットが形成されているから、小ドットは順序値「4」が設定された画素に形成される。図26では、小ドットが形成される画素には、粗い斜線を付して表示している。そして、最後に、大ドット、中ドット、小ドットのいずれも形成されなかつた画素については、ドットを形成しないものと判断する（図24のステップS614）。

【0156】

こうして、1つの画素群について、コード化された個数データを復号し、大中小の各ドットを形成する画素を決定したら、全画素群について処理を終了したか否かを判断する（図24のステップS616）。そして、未処理の画素群が残っている場合は（ステップS616：no）、ステップS600に戻って、新たな画素群について続く一連の処理を繰り返す。こうして全ての画素群について処理を終了したと判断されたら（ステップS616：yes）、図24に示したドット形成有無決定処理を終了する。

【0157】

C-3-2. 変換テーブルを参照するドット形成有無決定処理：

以上に説明したドット形成有無の決定処理では、コード化された個数データを受け取ると、画素群内に形成する大中小ドットの個数を示すデータに復号した後、各画素についていずれのドットを形成するかを決定するという、言わば2段階の操作を経ている。しかし、変換テーブルを参照すれば、各画素に形成するドットを、個数データを復号することなく直ちに決定することができる。以下では、変換テーブルを参照してドット形成の有無を決定する第2実施例のドット形成有無決定処理について説明する。

【0158】

第2実施例のドット形成有無決定処理は、前述した第1実施例において図14を用いて説明した変形例のドット形成有無決定処理に対して、参照する変換テーブルが異なっているだけであり、処理の流れは同様である。そこで、以下では、図14のフローチャートを流用しながら、第2実施例のドット形成有無決定処理について説明する。

【0159】

第2実施例のドット形成有無決定処理を開始すると、先ず初めに画素群を1つ選択し（ステップS400相当）、その画素群の個数データを取得する（ステップS402相当）。次いで、複数組記憶されている順序値マトリックスの中から、選択した画素群に対応する順序値マトリックスを読み込んでやる（ステップS404相当）。すなわち、図15および図16を用いて前述したように、画素群の座標値（n, m）から、nおよびmの下位5ビットを抜き出して、それぞれN, Mを求める。そして、ディザマトリックス中でN行M列のブロックから生成された順序値マトリックスを選択して、読み込んでやればよい。

【0160】

以上のようにして、画素群に対応する順序値マトリックスを読み込んだら、処理中の画素群の中から、ドット形成の有無を決定しようとする対象画素を1つ選択する（ステップS406相当）。そして、読み込んでおいた順序値マトリックスを参照することで、対象画素の順序値を取得した後（ステップS408相当）、変換テーブルを参照することによって、対象画素についてのドット形成の有無を決定する（ステップS410相当）。

【0161】

ここで、前述した第1実施例のドット形成有無決定処理において参照した変換テーブルには、個数データと順序値との組合せ毎に、ドット形成の有無を表すデータが設定されていた（図13参照のこと）。これに対して、第2実施例のドット形成有無決定処理で参照する変換テーブルには、コード化された状態の個数データと順序値との組合せ毎に、大中小いずれのドットを形成するか、あるいはドットを形成しないかを示すデータが設定されている。

【0162】

図27は、第2実施例のドット形成有無決定処理で参照される変換テーブルを概念的に示した説明図である。図23を用いて前述したように、コード化された個数データは0～164までの165通りの値を取り得る。また、1つの画素群は8つの画素から構成されているとしているから、順序値は1～8までの8通りの値を取り得る。第2実施例の変換テーブルには、これら $165 \times 8 = 1320$ 通りの組合せ毎に、ドットを形成しないことを示す値「0」、小ドットを形成することを示す値「1」、中ドットを形成することを示す値「2」、大ドットを形成することを示す値「3」のいずれかが設定されている。従つて、画素群の個数データと対象画素の順序値とが分かれれば、直ちにドット形成の有無を決定することができる。第2実施例のドット形成有無決定処理では、このような変換テーブルを参照することで、対象画素に大中小いずれのドットを形成するのか、あるいはドットを形成しないのかを、直ちに決定するのである（図14のステップS410相当）。

【0163】

ここで、図27に示すような変換テーブルを参照しながらドット形成の有無を決定した場合でも、大中小ドットについてのドット形成の有無を適切に決定可能な理由について説明する。図24ないし図26を用いて前述したように、変換テーブルを参照することなくドット形成の有無を決定する場合は、大きくは2段階の操作を経ている。すなわち、先ず第1段階では、コード化された個数データを大中小の各種ドットの個数に変換する。続く第2段階では、順序値マトリックスに従って各画素についてのドット形成の有無を決定する。ここで、図25に示されているように、コード化された個数データと、大中小ドットについての個数の組合せとは、一対一の関係にある。換言すれば、コード化された個数データが1つ与えられれば、各種ドットについての個数の組合せは一意的に決定することができる。

【0164】

一方、画素群内に形成される各種ドットの個数が復号されると、図26に示したように、順序値マトリックスに従って、各画素についてのドット形成の有無が決定される。すなわち、順序値マトリックスが決まつていれば、各種ドットについての個数の組合せと、各画素についてのドット形成の有無とは、一対一の関係にある。上述したように、各種ドットについての個数の組合せは、コード化された個数データから一意的に決定されるから、結局、順序値マトリックスが決まつていれば、コード化された個数データから、画素群内の各画素について、各種ドットの形成有無は一意的に決定されることになる。

【0165】

ここで、各種ドットの個数と順序値マトリックスとに従って、各画素についてのドット形成の有無を決定する操作を更に詳しく観察すると、次のようなことが分かる。すなわち、各種ドットの個数の組合せは、ある順序値の画素にはどのようなドットが形成されるかを決定しており、順序値マトリックスは、そのような順序値の画素が、画素群中のどの位置にあるかを決定している。例えば、図26に示した例では、大ドット1個、中ドット2個、小ドット1個であるから、順序値「1」の画素には大ドットが形成され、順序値「2」および「3」の画素には中ドットが形成され、順序値「4」の画素には小ドットが形成され、順序値「5」～「8」の画素にはいずれのドットも形成されないことがわかる。そして、これら順序値の画素が、画素群中ではどの位置にあるかについては、順序値マトリックスによって決定されている。このことから、コード化された個数データが与えられれば、ある順序値の画素にはどのようなドットが形成されるかについてまでは、一意的に決定することが可能である。

【0166】

そこで、全ての個数データについて、順序値とその順序値を有する画素に形成されるドットの種類とを予め決定しておく、図27に示すような変換テーブルに設定しておく。そして、画素群内の対象画素についてドット形成の有無を決定する場合には、順序値マトリックスを参照することで対象画素の順序値を求め、次いで変換テーブルを参照することで、その順序値に形成されるドットの種類を取得すれば、ドット形成の有無を適切に決定す

ることが可能となるのである。

【0167】

こうして、1つの対象画素についてドット形成の有無を決定したら、選択した画素群内の全画素について、ドット形成の有無を決定したか否かを判断する（図14のステップS412相当）。未だ、ドット形成の有無を決定していない画素が残っていれば、ステップS406に相当する箇所まで戻って、新たな対象画素を選択し、続く一連の処理を行う。こうした処理を繰り返し、画素群内の全画素についてドット形成の有無を決定したと判断されたら、今度は、個数データが供給された全画素群について処理を終了したか否かを判断する（ステップS414相当）。そして、未処理の画素群が残っていれば、一番初めまで戻って新たな画素群を選択し、続く一連の処理を行う。こうした処理を繰り返し、全ての画素群について処理が終了したら、第2実施例のドット形成有無決定処理を終了する。

【0168】

以上に説明した第2実施例のドット形成有無決定処理においては、コード化された状態の個数データを受け取ると、順序値マトリックスを参照して対象画素の順序値を取得した後、個数データと順序値とから変換テーブルを参照することで、個数データを復号することなく各種ドットについての形成有無を決定することができる。このため、画素毎にドット形成有無を決定する処理を迅速に実行することが可能となり、延いては画像を迅速に出力することが可能となる。

【0169】

また、第2実施例のドット形成有無決定処理の主な処理内容は、順序値マトリックスや変換テーブルを参照してデータを読み出しているだけであり、極めて単純な処理である。従って、コンピュータ100のように高いデータ処理能力を備えていないカラープリンタ200においても、十分に迅速に実行することが可能であり、それだけ画像を迅速に印刷することが可能となる。

【0170】

更に、第2実施例のドット形成有無決定処理では、マトリックスあるいはテーブルを参照するだけで対象画素についてのドット形成の有無を決定することが可能であり、処理中に条件分岐が含まれていない。これに対して、変換テーブルを参照することなくドット形成の有無を判断する場合は、多数の条件分岐が発生する。この点について図24を参照しながら若干の補足説明を行う。

【0171】

変換テーブルを参照せずに各種ドットの形成の有無を決定する場合、先ず初めに大ドットについての形成有無を判断する（図24のステップS608）。そして、大ドットが形成されなければ、中ドットの形成有無を判断する（図24のステップS610）。大ドットも中ドットも形成されなければ、今度は小ドットの形成有無を判断する（図24のステップS612）。そして、小ドットも形成されないと判断された画素については、いずれのドットも形成されないと判断している。このように、大ドットが形成されるか否か、中ドットが形成されるか否か、小ドットが形成されるか否かによって、以降の処理内容が異なっており、従って、ドット形成有無を決定する処理には、多数の条件分岐が含まれている。

【0172】

近年のCPUでは高速な処理を可能とするために、パイプライン処理と呼ばれる技術が採用されているが、条件分岐が発生すると、パイプライン処理による効果を得ることができない。この点で、変換テーブルを参照することなくドット形成有無を決定する処理は、上述したように多数の条件分岐を含んでいるために、迅速な処理を行うことが困難である。これに対して、変換テーブルを参照してドット形成有無を決定する処理は、図14に示したように、条件分岐を含まないためにパイプライン処理の効果を最大限に引き出すことが可能であり、この点でも高速処理に適した処理と言えることができる。

【0173】

以上に説明したように、第2実施例のドット形成有無決定処理では、画素群内の各画素

についてのドット形成有無を、簡便に且つ迅速に決定することができるが、多数の順序値マトリックスと図27に示したような変換テーブルを記憶しておく必要がある。これらマトリックスおよびテーブルを記憶するために、あまりに多くのメモリを使用したのでは、実際の製品に搭載する上で好ましいことではない。しかし、第2実施例のドット形成有無決定処理においても、前述した第1実施例と同様に、変換テーブルおよび順序値マトリックスを記憶するために必要なメモリ量は、製品に搭載する際の障害になるほど大きくなることはない。以下、この点について簡単に説明しておく。

【0174】

先ず、序列値マトリックスについては、前述した第1実施例と同様である。すなわち、順序値マトリックスを記憶するためのメモリ量は、ディザマトリックスの大きさと、画素群の大きさとによって決定され、図18に示したように、10Kバイト程度のメモリ量があれば、十分記憶することが可能と考えられる。

【0175】

次に、変換テーブルを記憶するために必要なメモリ容量について説明する。図27に示したように、変換テーブルには、順序値および個数データの組合せ毎にドット形成の有無が設定されている。画素群に含まれる画素数を8個とすれば、順序値は8通りの値を取る。また、形成可能なドットの種類が大ドット、中ドット、小ドットの3種類であるとすれば、図23を用いて前述したように、個数データは165通りの値を取り得る。更に、ドット形成有無の判断結果は、大ドットを形成する、中ドットを形成する、小ドットを形成する、ドットを形成しないの4通りしか取り得ないから、1つの判断結果は2ビットあれば表現することができる。従って、図27に示した変換テーブルは、 $8 \times 165 \times 2 = 2640$ ビット (0.322Kバイト) あれば記憶することができる。

【0176】

図28は、変換テーブルを記憶するために必要なメモリ量を、画素群の大きさ毎に試算した結果をまとめた説明図である。図示されているように、変換テーブルについても、数Kバイト程度あれば、十分記憶することができる。従って、第2実施例のドット形成有無決定処理においても、順序値マトリックスや変換テーブルを記憶するためには僅かなメモリ量があればよく、メモリ容量が製品への搭載の障害となることはない。

【0177】

以上、各種の実施例について説明してきたが、本発明は上記すべての実施例に限られるものではなく、その要旨を逸脱しない範囲において種々の態様で実施することができる。例えば、以上の実施例では、印刷用紙上にドットを形成して画像を印刷する場合について説明したが、本発明の適用範囲は画像を印刷する場合に限られるものではない。例えば、液晶表示画面上で輝点を適切な密度で分散させることにより、階調が連続的に変化する画像を表現する液晶表示装置などにも、本発明を好適に適用することができる。

【図面の簡単な説明】

【0178】

【図1】印刷システムを例にとって本発明の概要を説明するための説明図である。

【図2】画像処理装置としてのコンピュータの構成を示す説明図である。

【図3】本実施例のカラープリンタ200の概略構成を示す説明図である。

【図4】インク吐出用ヘッドにおけるインクジェットノズルの配列を示す説明図である。

【図5】第1実施例の画像印刷処理の全体的な流れを示すフローチャートである。

【図6】ディザマトリックスの一部を概念的に例示した説明図である。

【図7】ディザマトリックスを参照しながら各画素についてのドット形成の有無を判断している様子を概念的に示した説明図である。

【図8】ディザ法を用いて画像データをドット形成の有無を表すデータに変換している様子を示した説明図である。

【図9】個数データから画素毎にドット形成の有無を表すデータを生成する様子を示した説明図である。

【図10】第1実施例の個数データ生成処理の流れを示すフローチャートである。

【図11】第1実施例のドット形成有無決定処理の流れを示すフローチャートである。

【図12】第1実施例のドット形成有無決定処理において各画素についてのドット形成の有無が決定される様子を概念的に示した説明図である。

【図13】対象画素についてのドット形成の有無を決定するために参照される変換テーブルを概念的に示した説明図である。

【図14】変形例のドット形成有無決定処理の流れを示すフローチャートである。

【図15】変形例のドット形成有無決定処理において参照される複数の順序値マトリックスを生成する方法について示した説明図である。

【図16】画素群に対応する順序値マトリックスを選択する方法を示した説明図である。

【図17】画素群の座標値 (n, m) から適用する順序値マトリックスを選択する方法を具体的に示した説明図である。

【図18】各種大きさのディザマトリックスと各種大きさの画素群とを想定して順序値マトリックスを記憶するために必要なメモリ量を試算した結果を示す説明図である。

【図19】画素群内に形成される大中小の各種ドットの個数を決定して個数データを生成する処理の流れを示すフローチャートである。

【図20】選択した画素について大中小の各種ドットについての形成有無を判断する処理の流れを示すフローチャートである。

【図21】画像データの階調値を大中小の各種ドットについての密度データに変換するためのドット密度変換テーブルを概念的に示した説明図である。

【図22】画素群内の各画素についてディザ法を適用しながら大中小の各種ドットの形成有無を判断している様子を概念的に示した説明図である。

【図23】画素群に形成される大中小各ドットの個数の組合せとコードデータとが対応付けて設定された対応テーブルを示す説明図である。

【図24】変換テーブルを参照することなく大中小の各種ドットの形成有無を決定する処理の流れを示すフローチャートである。

【図25】コード化された個数データを復号するために参照される復号テーブルを概念的に示した説明図である。

【図26】順序値マトリックスを参照しながら、大中小の各種ドットについての形成有無を画素毎に決定している様子を概念的に示した説明図である。

【図27】第2実施例のドット形成有無決定処理で参照される変換テーブルを概念的に示した説明図である。

【図28】変換テーブルを記憶するために必要なメモリ量を画素群の大きさ毎に試算した結果をまとめた説明図である。

【符号の説明】

【0179】

10…コンピュータ、 20…カラープリンタ、 100…コンピュータ、

108…周辺機器インターフェースP I F、

109…ディスクコントローラD D C、

110…ネットワークインターフェースカードN I C、

112…ビデオインターフェースV I F、

116…バス、 118…ハードディスク、 120…デジタルカメラ、

122…カラースキャナ、 124…フレキシブルディスク、

126…コンパクトディスク、 230…キャリッジモータ、 235…モータ、

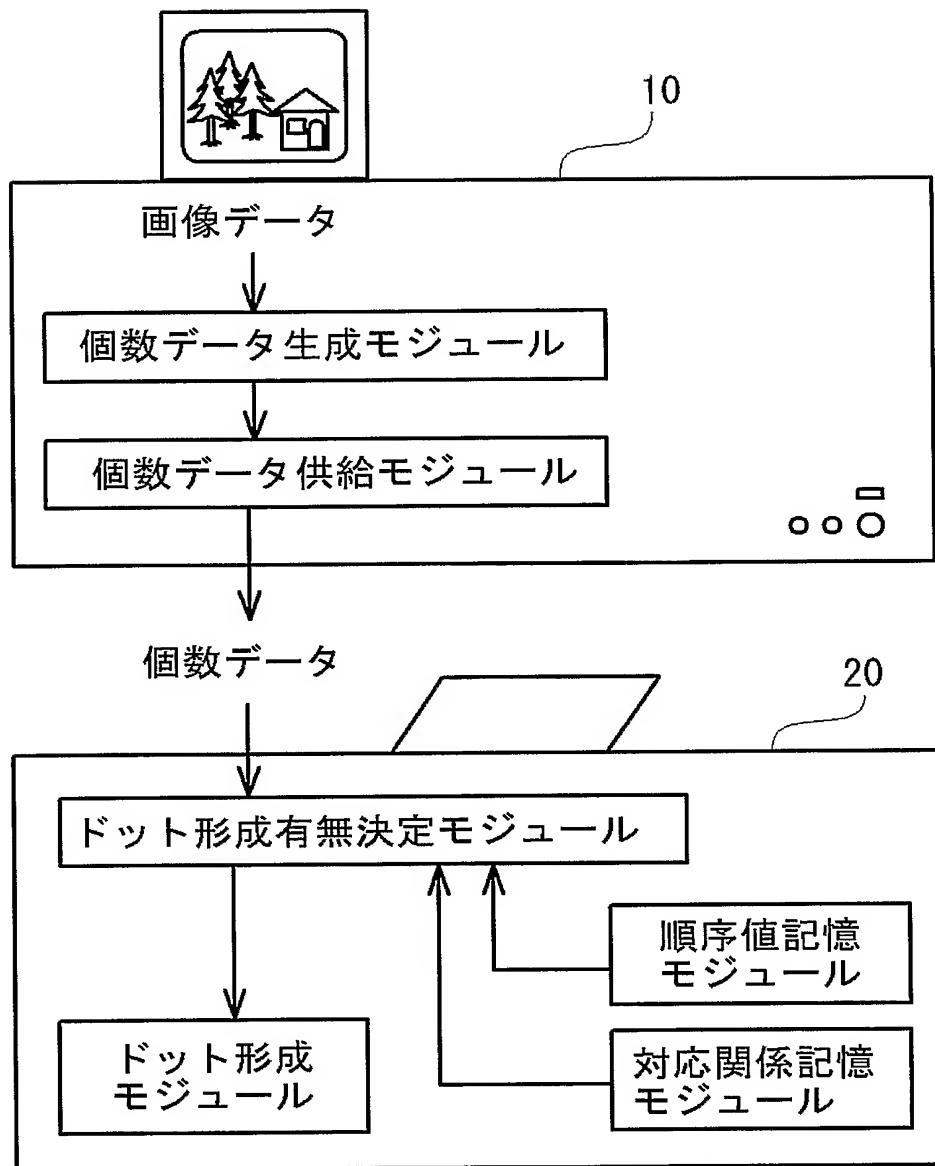
236…プラテン、 240…キャリッジ、 241…印字ヘッド、

242…インクカートリッジ、 243…インクカートリッジ、

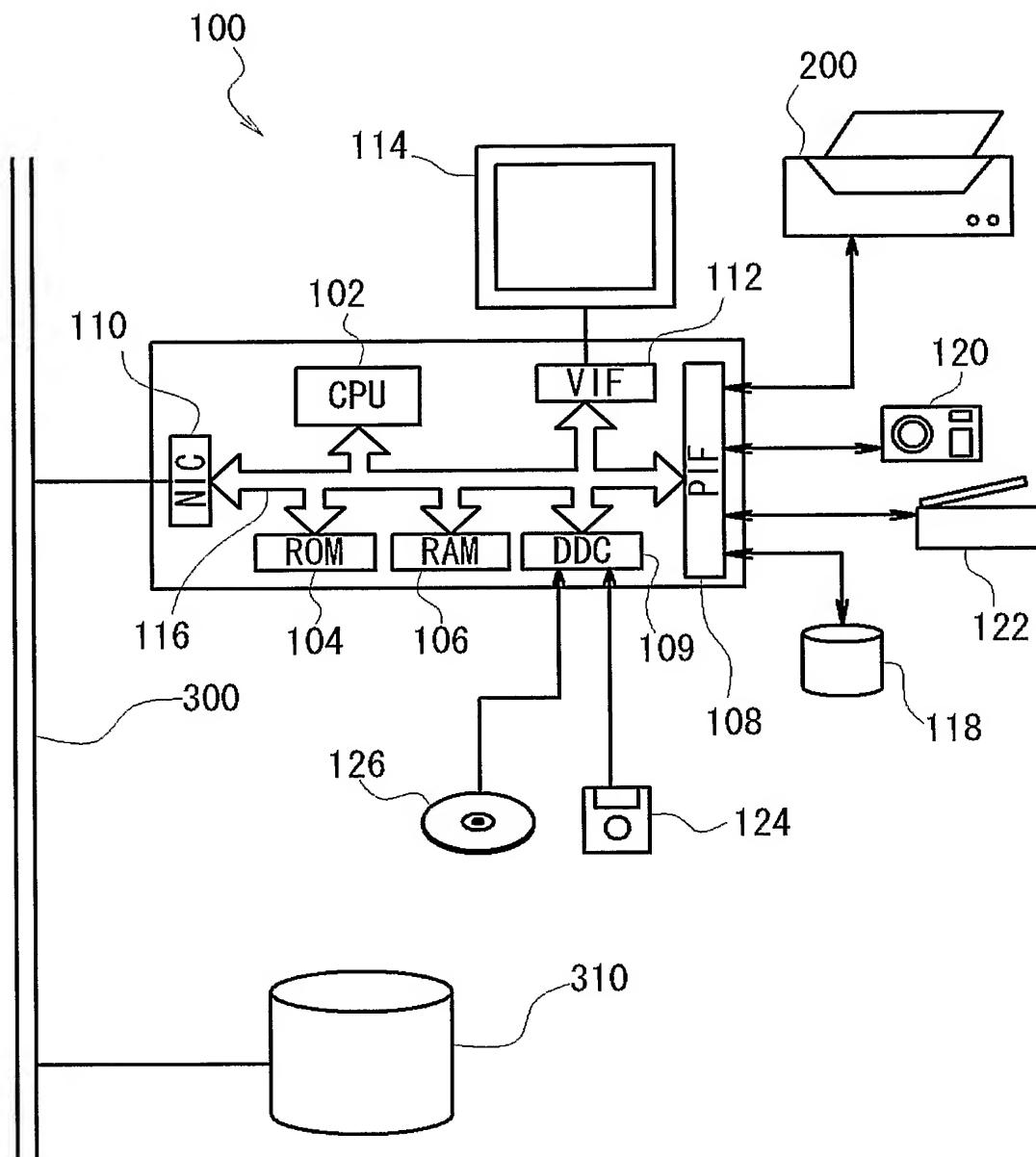
244…インク吐出用ヘッド、 260…制御回路、 200…カラープリンタ、

300…通信回線、 310…記憶装置

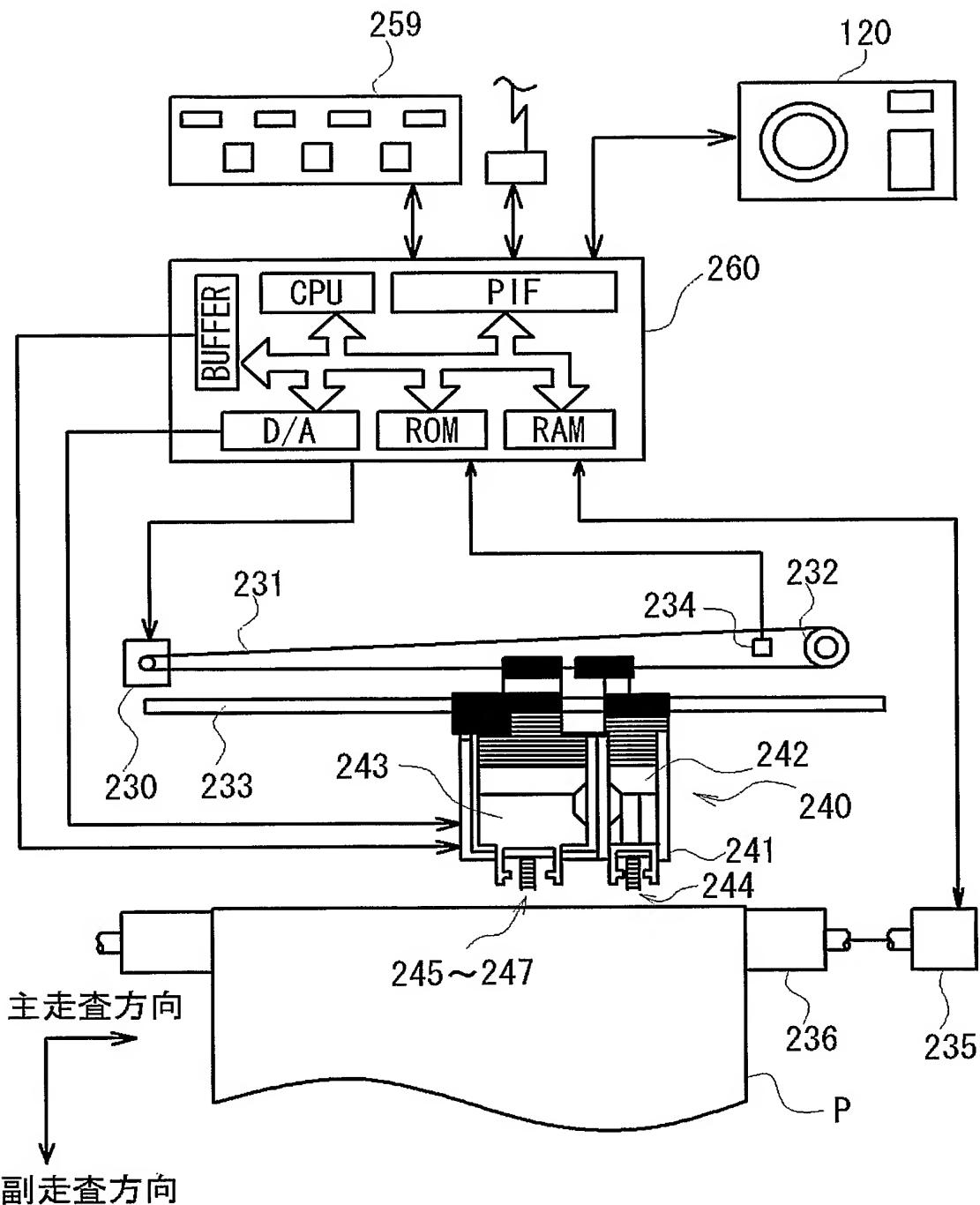
【書類名】図面
【図1】



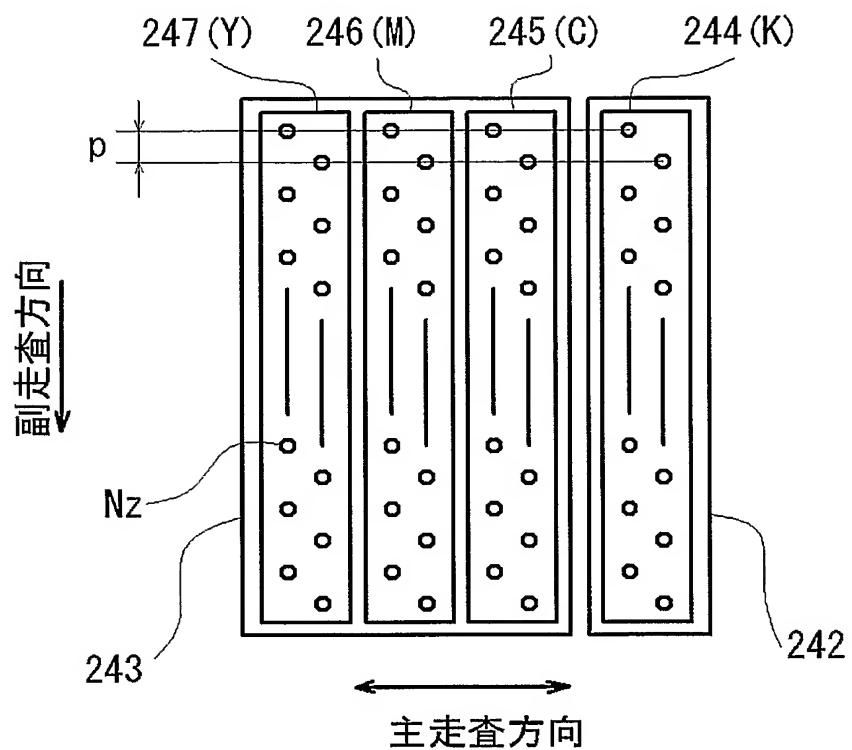
【図2】



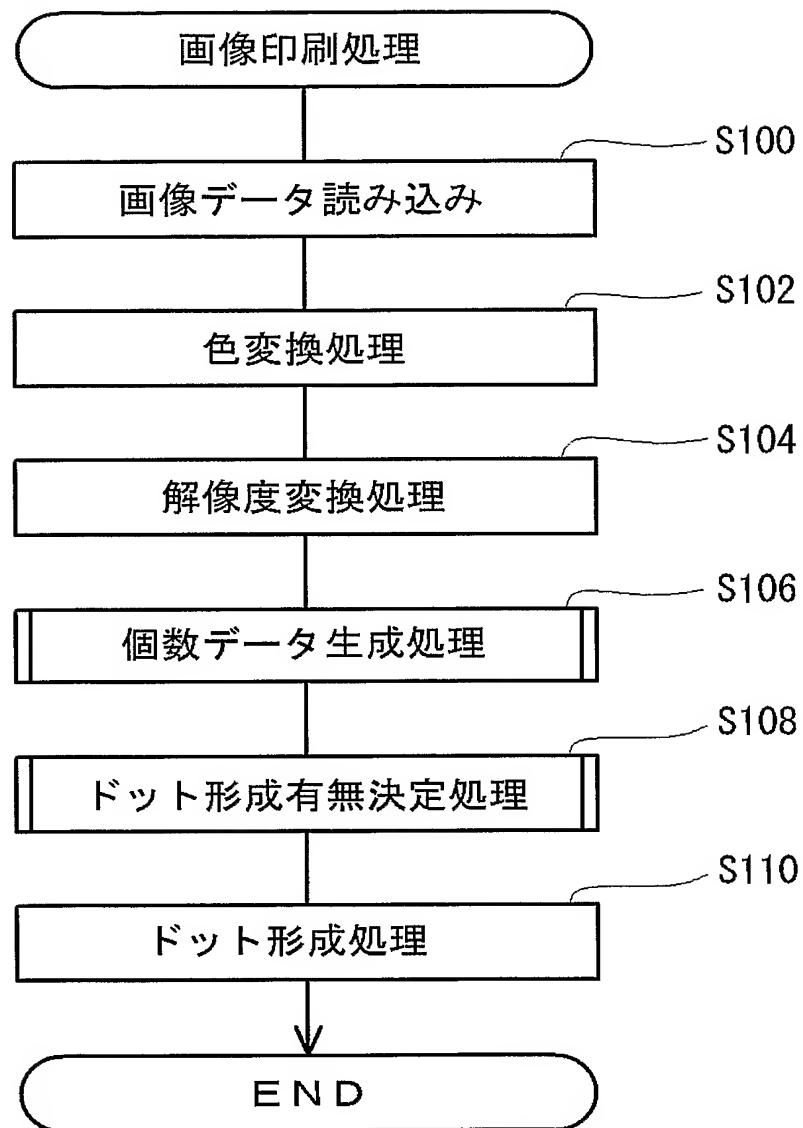
【図 3】



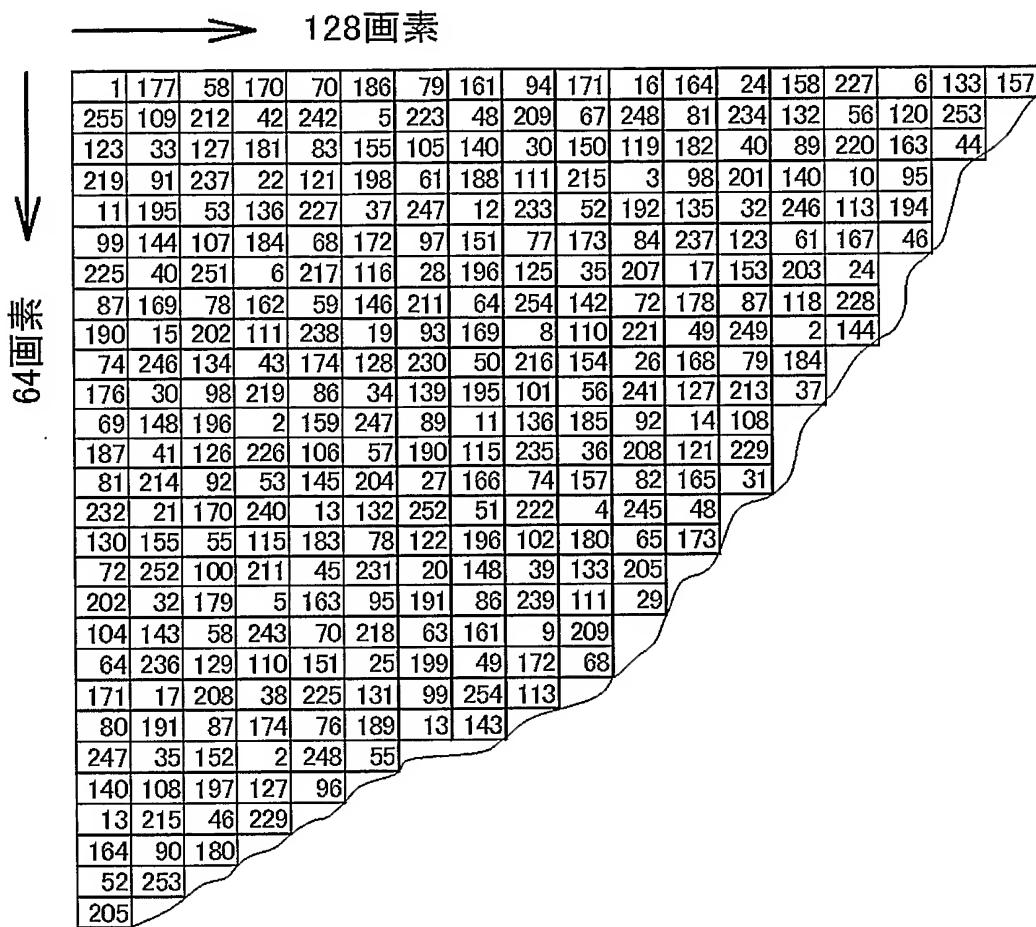
【図 4】



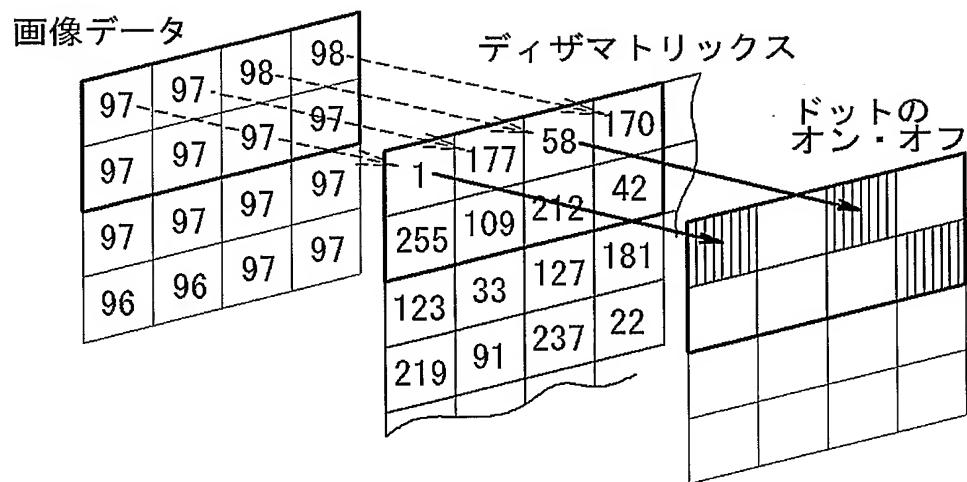
【図5】



【図 6】



【図7】



【図 8】

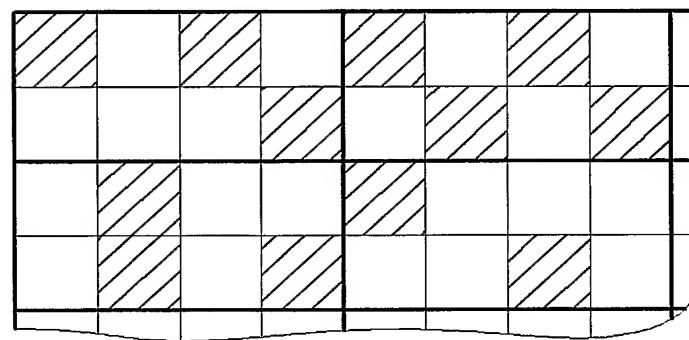
(a)

97	97	98	98	98	98	98	98	
97	97	97	97	97	97	98	98	
97	97	97	97	97	97	97	97	
96	96	97	97	97	97	97	97	

(b)

1	177	58	170	70	186	79	161	
255	109	212	42	242	5	223	48	
123	33	127	181	83	155	105	140	
219	91	237	22	121	198	61	188	

(c)

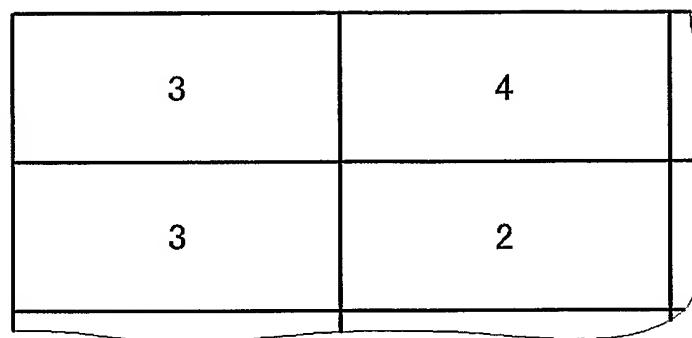


(d)

3	4
3	2

【図 9】

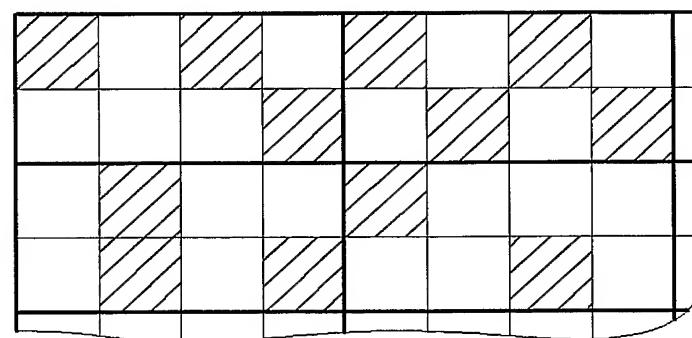
(a)



(b)

1	177	58	170	70	186	79	161
255	109	212	42	242	5	223	48
123	33	127	181	83	155	105	140
219	91	237	22	121	198	61	188

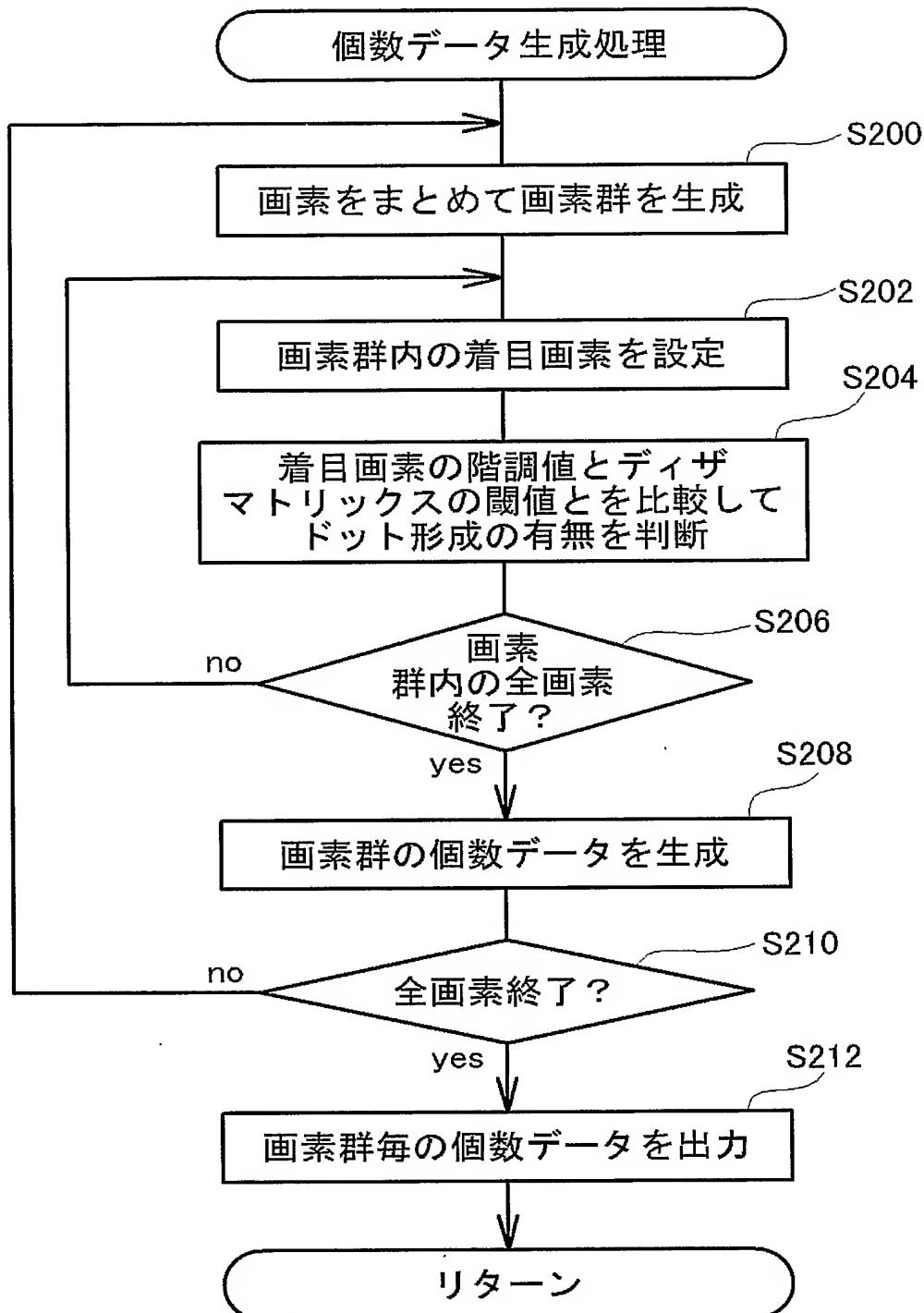
(c)



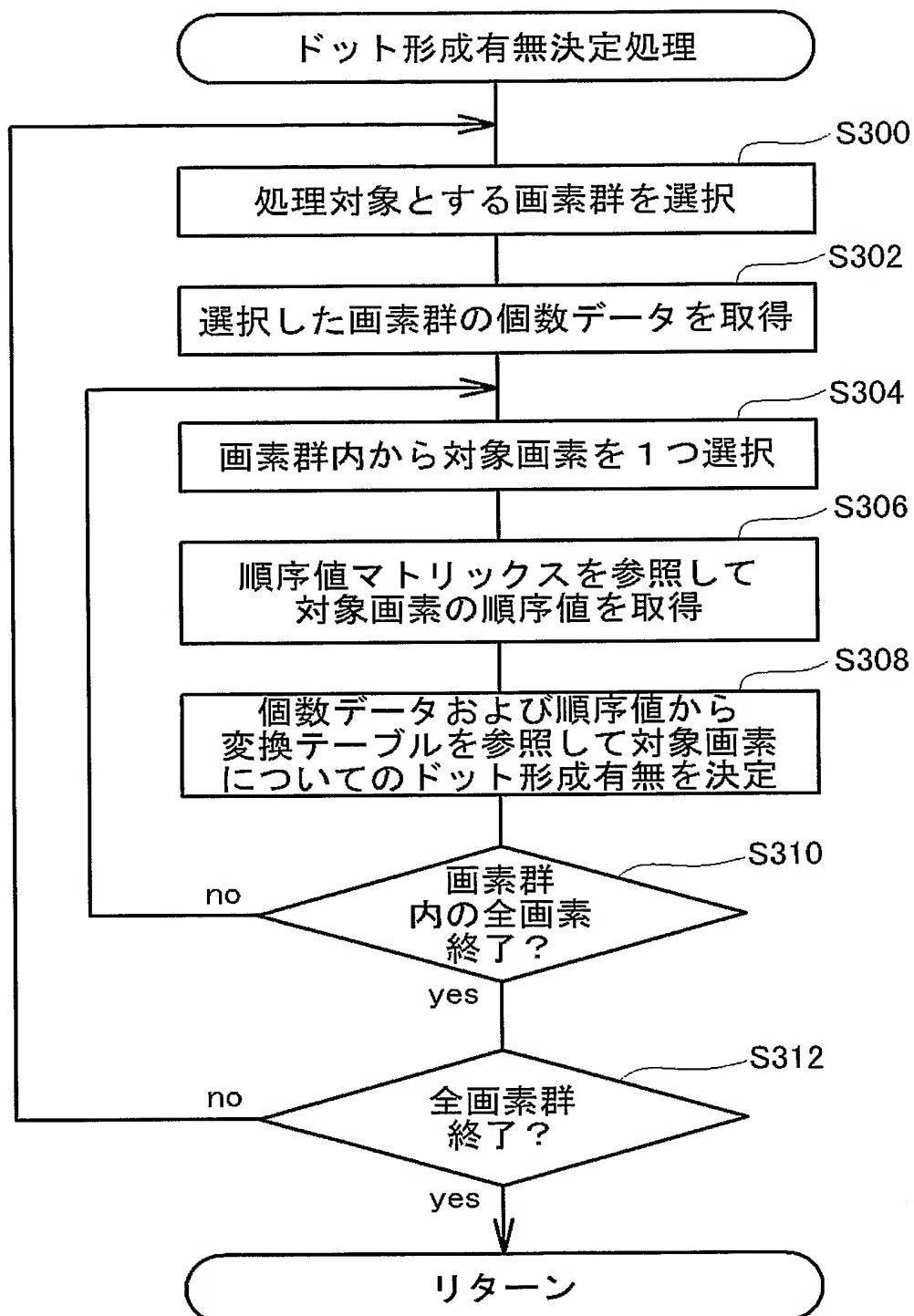
(d)

1	6	3	5	3	6	4	5
8	4	7	2	8	1	7	2
4	2	5	6	2	6	3	5
7	3	8	1	4	8	1	7

【図10】

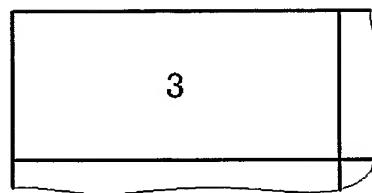


【図11】



【図 12】

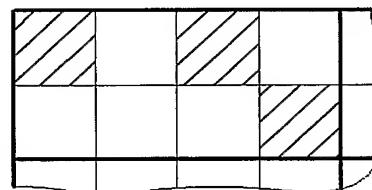
(a)



(b)

1	6	3	5
8	4	7	2

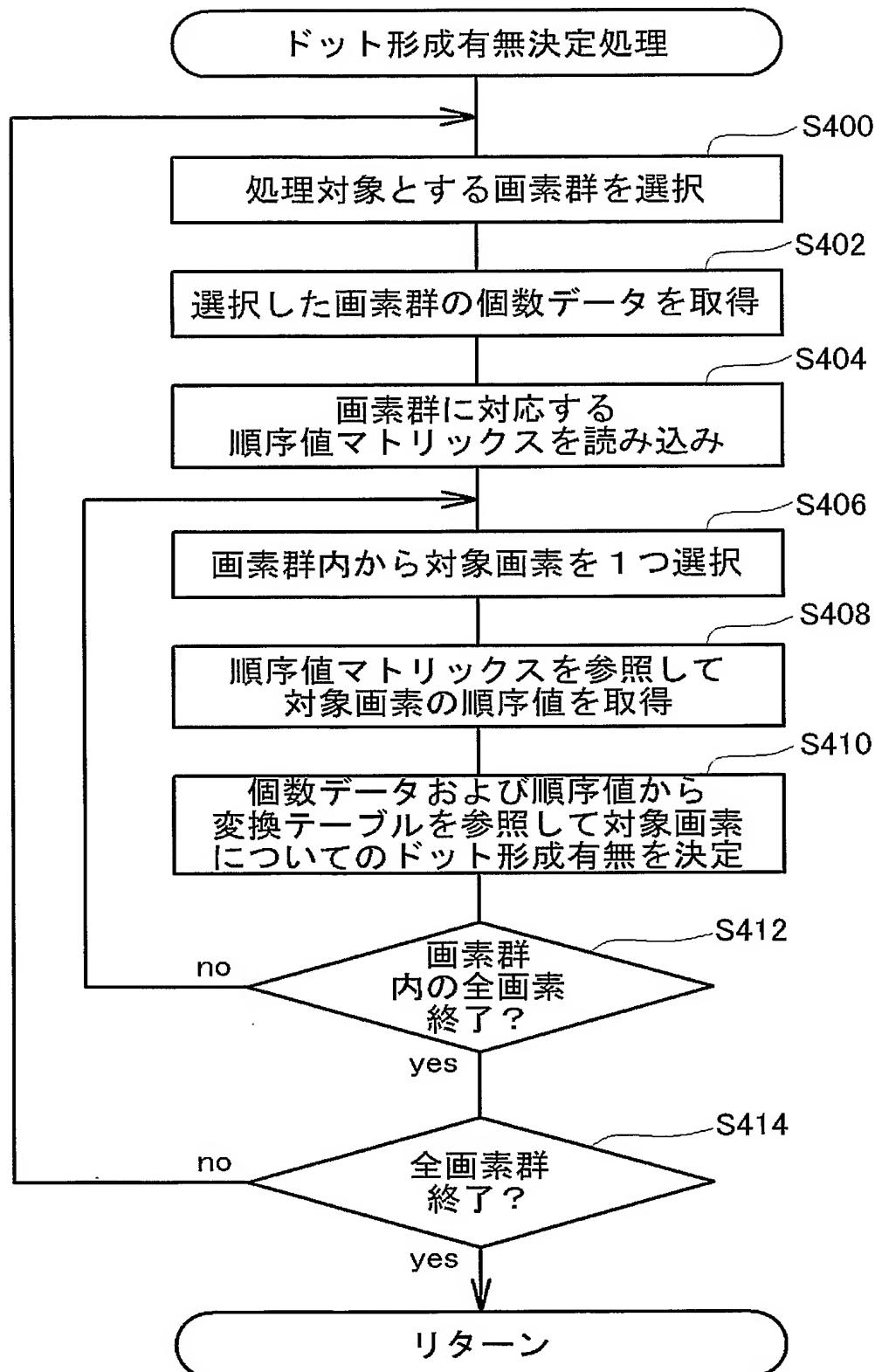
(c)



【図 13】

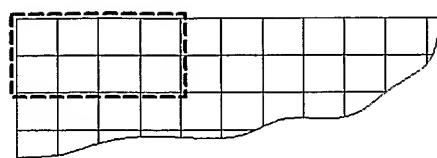
		個数データ								
		0	1	2	3	4	5	6	7	8
順序 値	1	0	1	1	1	1	1	1	1	1
	2	0	0	1	1	1	1	1	1	1
	3	0	0	0	1	1	1	1	1	1
	4	0	0	0	0	1	1	1	1	1
	5	0	0	0	0	0	1	1	1	1
	6	0	0	0	0	0	0	1	1	1
	7	0	0	0	0	0	0	0	1	1
	8	0	0	0	0	0	0	0	0	1

【図14】

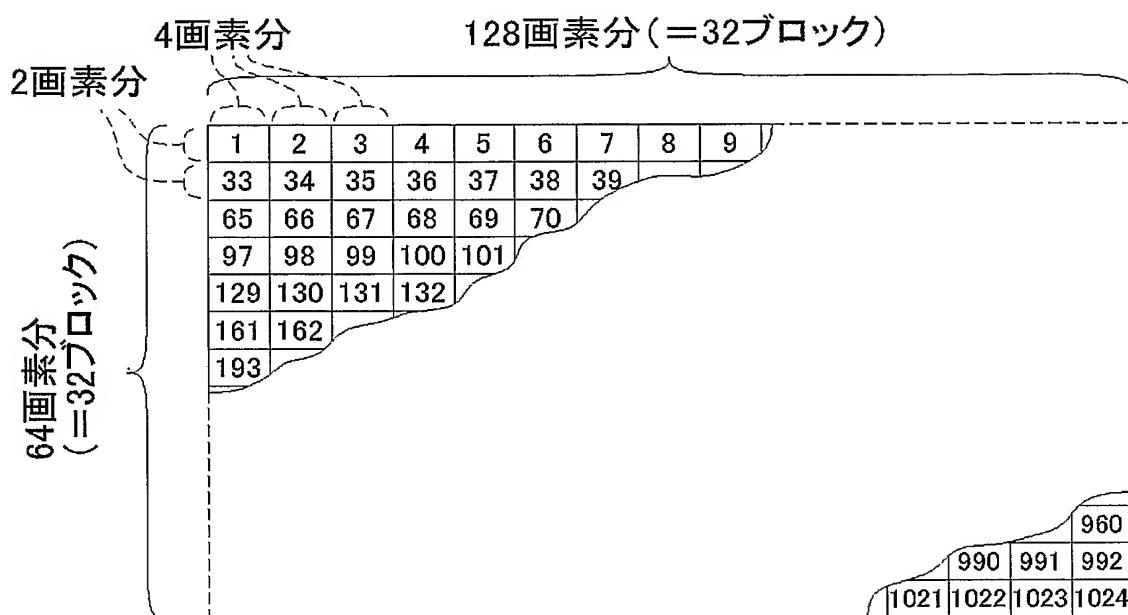


【図15】

(a)



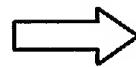
(b)



(c)

通し番号1番のブロック

1	177	58	170
255	109	212	42



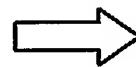
1番の順序値マトリックス

1	6	3	5
8	4	7	2

(d)

通し番号2番のブロック

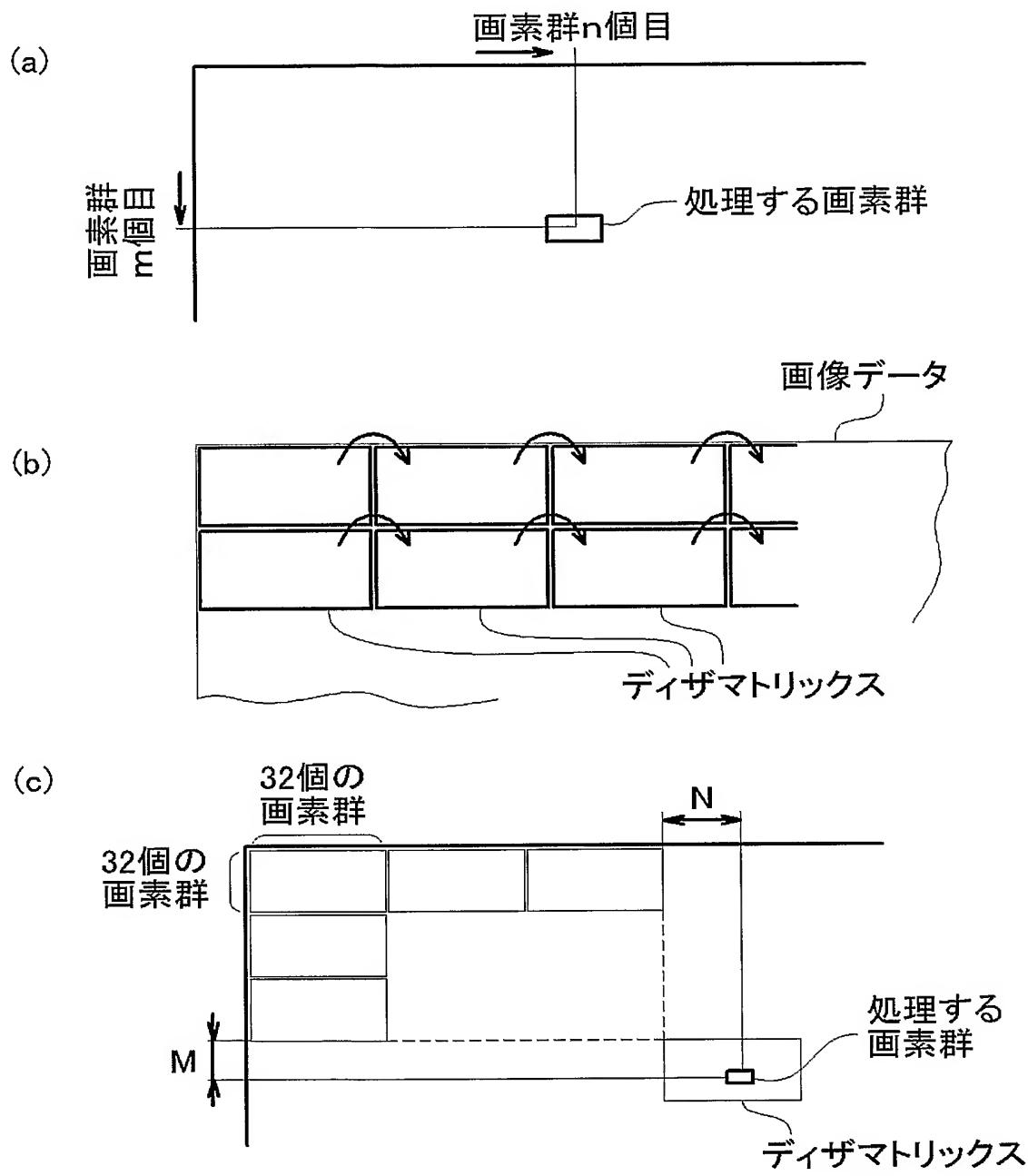
70	186	79	161
242	5	223	48



2番の順序値マトリックス

3	6	4	5
8	1	7	2

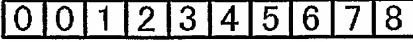
【図16】

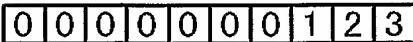


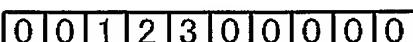
(d)

$$\begin{cases} N = n - \text{int}(n/32) \times 32 \\ M = m - \text{int}(m/32) \times 32 \end{cases}$$

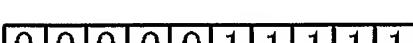
【図 17】

(a) n :  右方向に
5ビットシフト

(b) $\text{int}(n/32)$:  左方向に
5ビットシフト

(c) $\text{int}(n/32) \times 32$: 

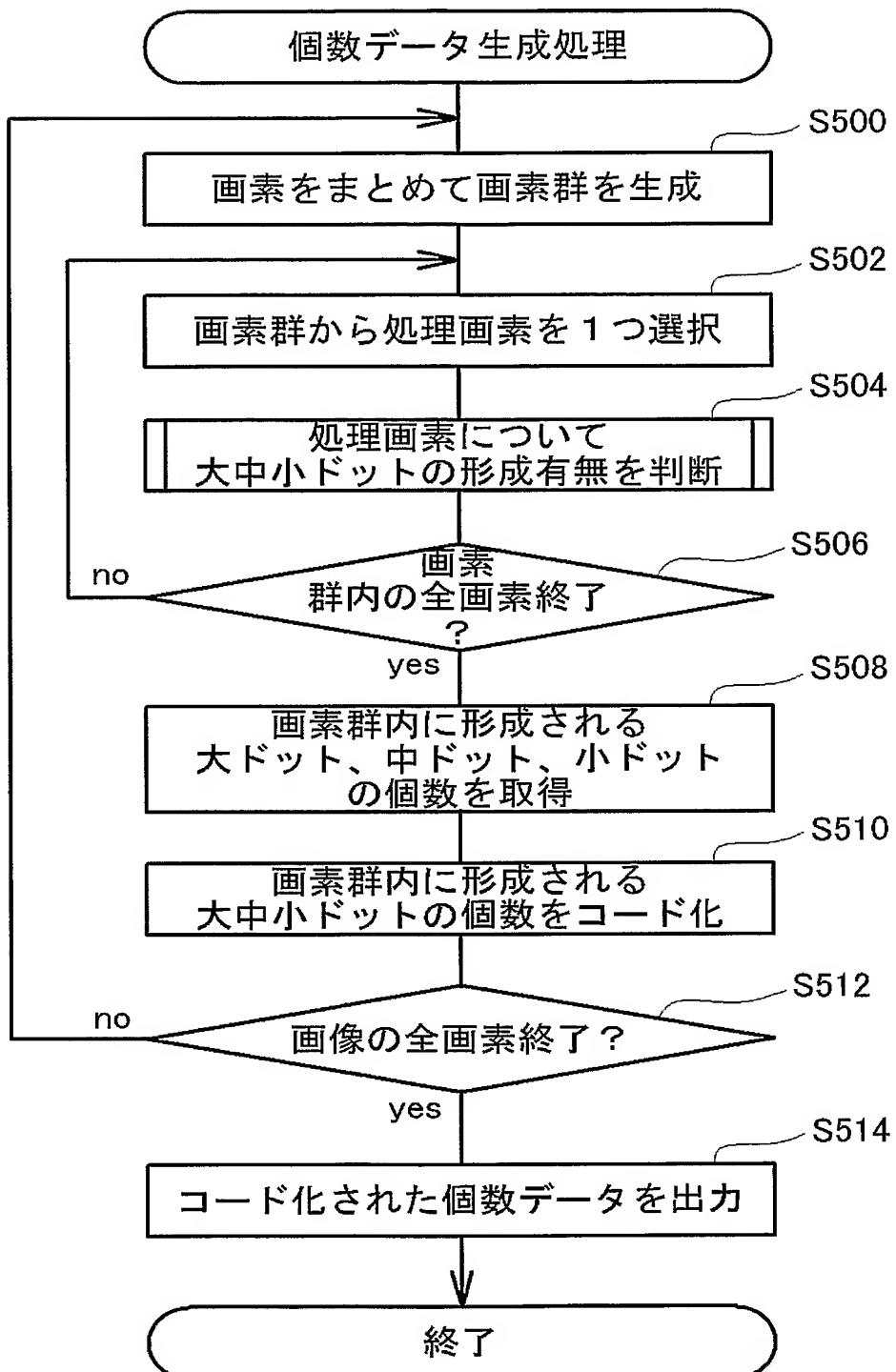
(d) $n - \text{int}(n/32) \times 32$: 

(e) マスクデータ : 

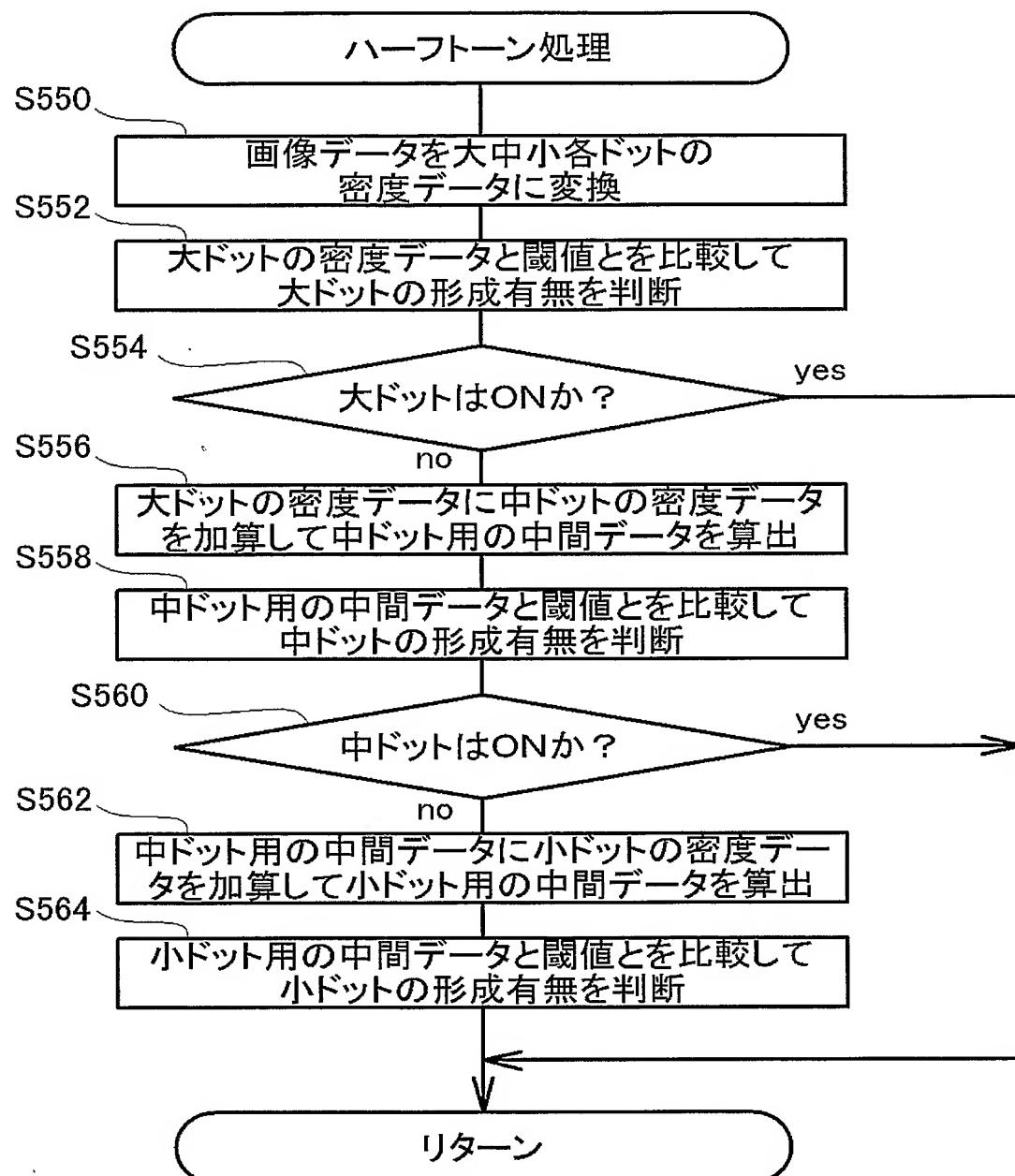
【図18】

ディザマトリックス サイズ (画素)		画素群サイズ (画素)		順序値 マトリックス の個数	順序値 マトリックス のサイズ (バイト)	必要な メモリ量 (Kバイト)
主走査 方向	副走査 方向	主走査 方向	副走査 方向			
64	64	2	2	1024	1	1
		4	2	512	3	1.5
		4	4	256	8	2
128	64	2	2	2048	1	2
		4	2	1024	3	3
		4	4	512	8	4
128	128	2	2	4096	1	4
		4	2	2048	3	6
		4	4	1024	8	8

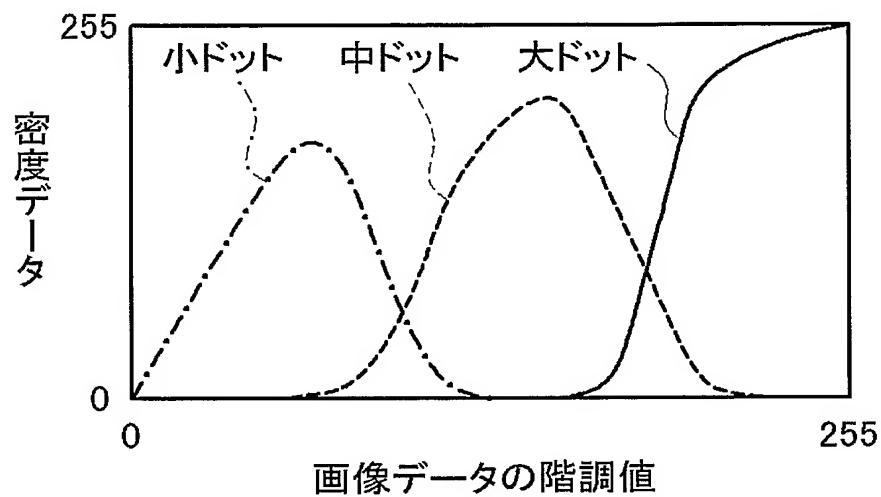
【図19】



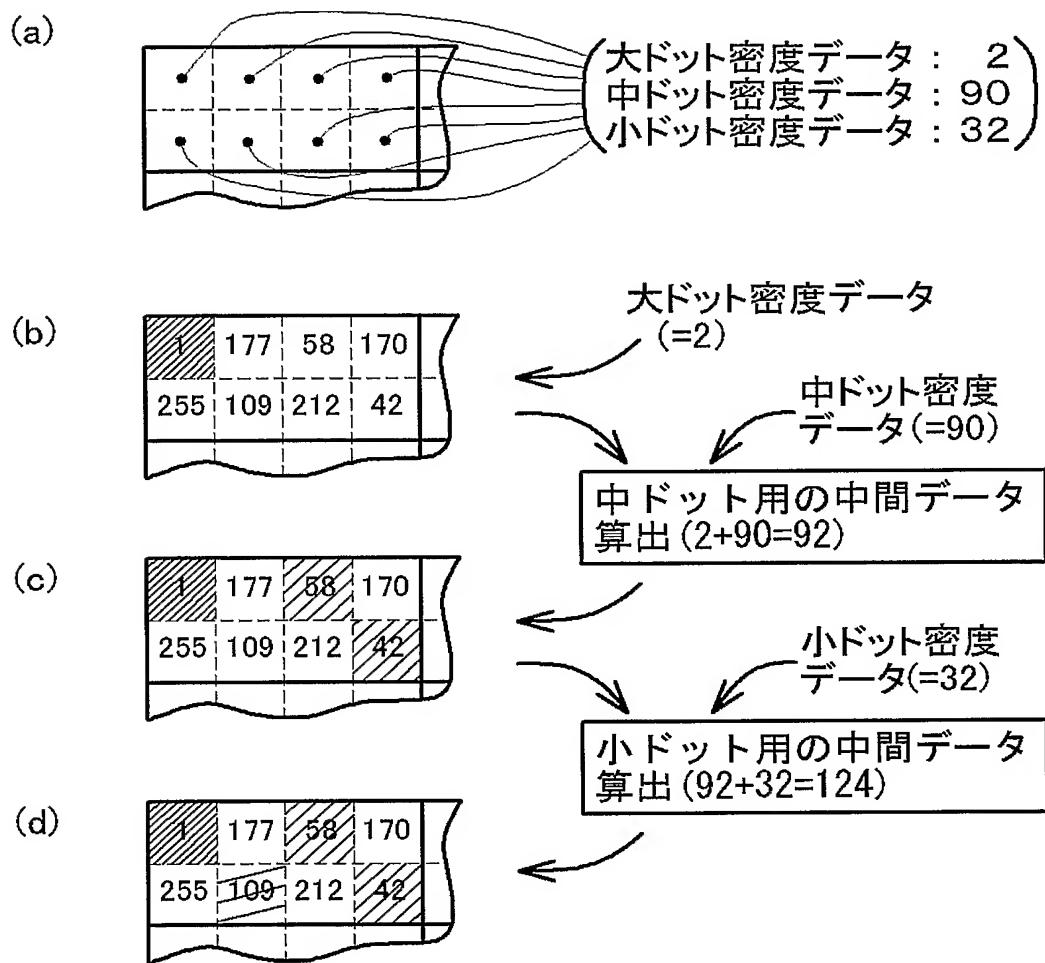
【図20】



【図21】



【図22】

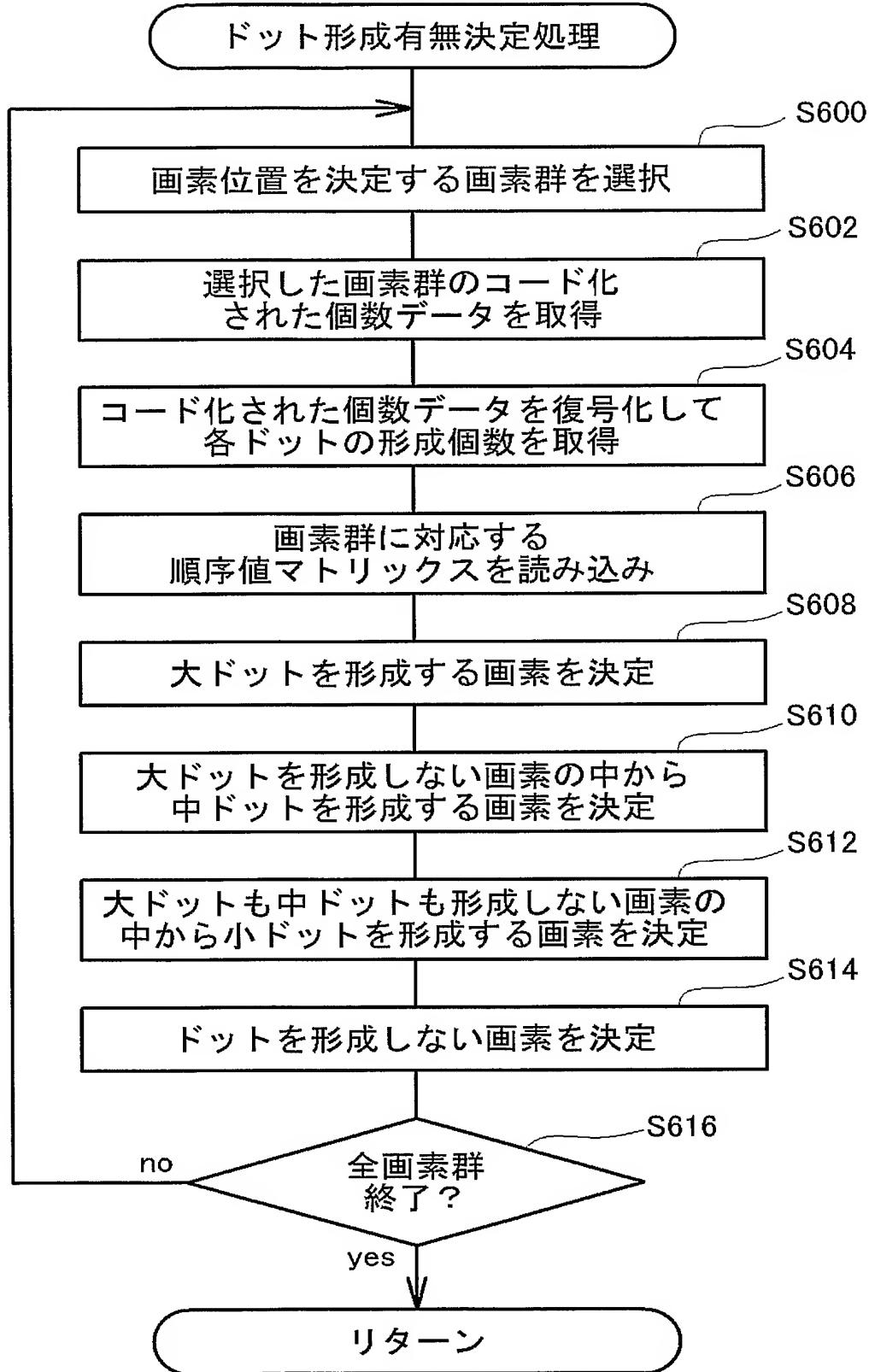


【図23】

ドット個数			コード化された 個数データ
大	中	小	
0	0	0	0
0	0	1	1
0	0	2	2
0	0	3	3

6	2	0	160
7	0	0	161
7	0	1	162
7	1	0	163
8	0	0	164

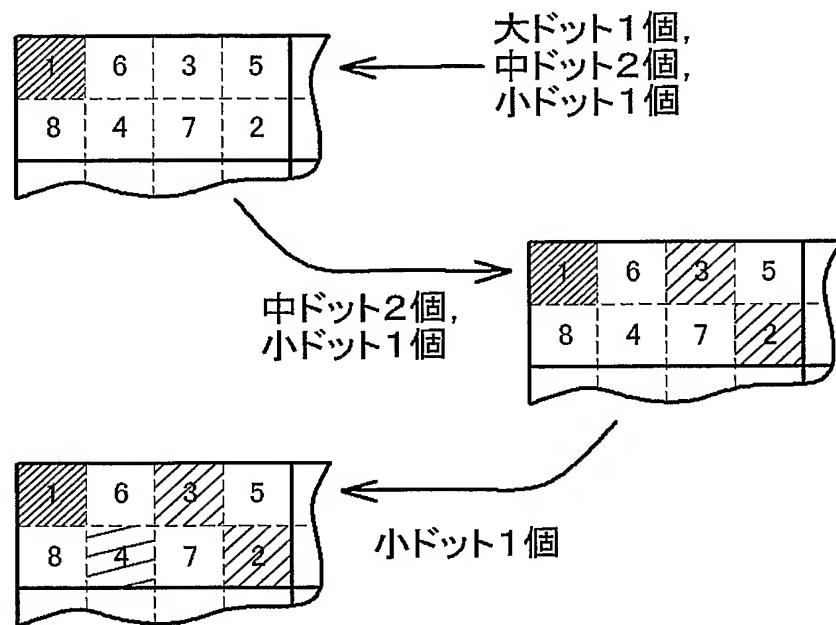
【図24】



【図 25】

コード化 された 個数データ	ドット個数		
	大	中	小
0	0	0	0
1	0	0	1
2	0	0	2
3	0	0	3
160	6	2	0
161	7	0	0
162	7	0	1
163	7	1	0
164	8	0	0

【図26】



【図 27】

		個数データ												
		0	1	2	3	4	5	6	7	162	163	164
順序値	1	0	1	1	1	1	1	1	162	163	164	
	2	0	0	1	1	1	1	1	3	3	3	
	3	0	0	0	1	1	1	1	3	3	3	
	4	0	0	0	0	1	1	1	3	3	3	
	5	0	0	0	0	0	0	0	3	3	3	
	6	0	0	0	0	0	0	0	3	3	3	
	7	0	0	0	0	0	0	0	3	3	3	
	8	0	0	0	0	0	0	0	3	3	3	

【図 28】

画素群サイズ (画素)		順序値 の個数	コード化さ れた個数 データの 個数	変換テー ブルに必要な メモリ量 (Kバイト)
主走査 方向	副走査 方向			
2	2	4	35	0.034
4	2	8	165	0.322
4	4	16	969	3.785

【書類名】要約書

【要約】

【課題】 高画質な画像を簡素な処理で迅速に出力可能な技術を提供する。

【解決手段】 隣接する画素を所定数ずつまとめて画像を複数の画素群に分割し、画素群内の各画素について、ドットが形成される順番を示す順序値を予め設定しておく。また、画素群内に形成されるドット個数のデータと順序値との組合せ毎に、該順序値を有する画素でのドット形成有無との対応関係を記憶しておく。画像の出力に際しては、画素群の個数データを生成し、個数データおよび順序値の組合せから対応関係を参照して各画素についてのドット形成の有無を決定し、得られた結果に従ってドットを形成する。個数データはデータ量が少ないため迅速にデータをやり取り可能であり、対応関係を参照すれば、各画素のドット形成有無を迅速に決定することができる、高画質な画像データを迅速に出力することが可能となる。

【選択図】

図1

認定・付加情報

特許出願の番号	特願2004-044649
受付番号	50400277620
書類名	特許願
担当官	第八担当上席 0097
作成日	平成16年 2月23日

<認定情報・付加情報>

【提出日】	平成16年 2月20日
-------	-------------

特願 2004-044649

出願人履歴情報

識別番号 [000002369]

1. 変更年月日 1990年 8月20日

[変更理由] 新規登録

住所 東京都新宿区西新宿2丁目4番1号
氏名 セイコーホームズ株式会社